



TUGAS AKHIR - SM141501

APLIKASI LEARNING VECTOR QUANTIZATION DAN PRINCIPAL COMPONENT ANALYSIS PADA PENGENALAN INDIVIDU MELALUI IDENTIFIKASI IRIS MATA

Melynda Sylvia Dewy
NRP 1213 100 070

Dosen Pembimbing:
Drs. Nurul Hidayat, M. Kom
Drs. Soetrisno, Ml. Kom

DEPARTEMEN MATEMATIKA
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember
Surabaya
2017



FINAL PROJECT - SM141501

APPLICATION OF LEARNING VECTOR QUANTIZATION AND PRINCIPAL COMPONENT ANALYSIS ON THE INDIVIDUAL RECOGNITION THROUGH IRIS IDENTIFICATION

Melynda Sylvia Dewy
NRP 1213 100 070

Supervisors:

Drs. Nurul Hidayat, M. Kom
Drs. Soetrisno, Ml. Kom

MATHEMATICS DEPARTMENT
Faculty of Mathematics and Natural Science
Sepuluh Nopember Institute of Technology
Surabaya
2017

LEMBAR PENGESAHAN

APLIKASI LEARNING VECTOR QUANTIZATION DAN PRINCIPAL COMPONENT ANALYSIS PADA PENGENALAN INDIVIDU MELALUI IDENTIFIKASI IRIS MATA

APPLICATION OF LEARNING VECTOR QUANTIZATION AND PRINCIPAL COMPONENT ANALYSIS ON THE INDIVIDUAL RECOGNITION THROUGH IRIS IDENTIFICATION

TUGAS AKHIR

Diajukan untuk Memenuhi Salah Satu Syarat untuk Memperoleh
Gelar Sarjana Sains pada Bidang Studi Ilmu Komputer
Program Studi S-1 Departemen Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Institut Teknologi Sepuluh Nopember Surabaya

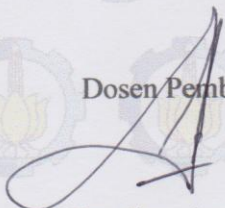
Oleh:

**MELYNDA SYLVIA DEWY
NRP. 1213 100 070**

Menyetujui,

Dosen Pembimbing II

Dosen Pembimbing I


Drs. Soetrisno, M.I.Kom

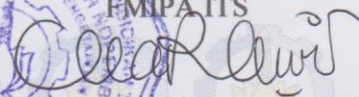
NIP. 19571103 198603 1 003


Drs. Nurul Hidayat, M.Kom

NIP. 19630404 198903 1 002

Mengetahui,

Kepala Departemen Matematika
FMIPA ITS


Dr. Imam Mukhlash, S.Si., M.T

NIP. 19700831 199403 1 003

Surabaya, Juli 2017

APLIKASI *LEARNING VECTOR QUANTIZATION* DAN *PRINCIPAL COMPONENT ANALYSIS* PADA PENGENALAN INDIVIDU MELALUI IDENTIFIKASI IRIS MATA

Nama : Melynda Sylvia Dewy
NRP : 1213 100 070
Jurusan : Matematika
Dosen Pembimbing : 1. Drs. Nurul Hidayat, M. Kom
2. Drs. Soetrisno, MI. Kom

Abstrak

Biometrika merupakan cabang matematika terapan yang memanfaatkan karakteristik biologis yang bersifat unik yang dimiliki oleh individu untuk proses identifikasi. Karakteristik unik yang ada pada tubuh individu yang dapat digunakan sebagai biometrika antara lain, seperti wajah, telapak tangan, sidik jari, suara, dan iris mata. Dalam penelitian ini dibuat sistem pengenalan individu melalui identifikasi iris mata menggunakan metode *Principal Component Analysis (PCA)* untuk mendapatkan ciri dari citra iris mata dan *Learning Vector Quantization (LVQ)* digunakan untuk proses pelatihan dan pengujian pada tahap klasifikasi. Data yang digunakan dalam penelitian ini adalah data citra mata dari 108 mata berbeda yang diambil dari *CASIA* database versi 1.0. Uji coba dilakukan pada 30, 50, 80 dan 108 mata yang menghasilkan akurasi sebesar 93.518% pada 108 individu dengan pengambilan 100 vektor eigen, *learning rate*=0.1, *epoch*=100 dan penurunan *learning rate*=0.1.

Kata Kunci: *biometrika, iris mata, Principal Component Analysis, Learning Vector Quantization*

**APPLICATION OF LEARNING VECTOR QUANTIZATION
AND PRINCIPAL COMPONENT ANALYSIS ON THE
INDIVIDUAL RECOGNITION THROUGH IRIS
IDENTIFICATION**

Name of Student : Melynda Sylvia Dewy
NRP : 1213 100 070
Department : Mathematics
Supervisors : 1. Drs. Nurul Hidayat, M. Kom
2. Drs. Soetrisno, MI. Kom

Abstract

Biometrics is a branch of applied mathematics that utilizes unique biological characteristics possessed by individuals for the identification process. The unique characteristics that exist on the individual body that can be used as biometrics include, such as face, palms, fingerprints, speech, and iris. In this study an individual recognition system was developed through the identification of the iris using Principal Component Analysis (PCA) method to obtain the features of the iris image and Learning Vector Quantization (LVQ) was used for the training and testing process at the classification stage. The data used in this research is eye image data from 108 different eyes taken from CASIA database version 1.0. Trials were performed on 30, 50, 80 and 108 eyes which resulted in an accuracy of 93.518% in 108 individuals with 75 eigen vectors, learning rate = 0.1, epoch = 100 and decreased learning rate = 0.1.

Keywords: *Biometrics, iris, Principal Component Analysis, Learning Vector Quantization*

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Alhamdulillahirobbil'aalamiin, segala puji dan syukur penulis panjatkan kepada Allah SWT atas limpahan rahmat, petunjuk dan hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul

“APLIKASI *LEARNING VECTOR QUANTIZATION* DAN *PRINCIPAL COMPONENT ANALYSIS* PADA PENGENALAN INDIVIDU MELALUI IDENTIFIKASI IRIS MATA”

sebagai salah satu syarat kelulusan Program Sarjana Jurusan Matematika FMIPA Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

Tugas Akhir ini dapat diselesaikan dengan baik berkat kerjasama, bantuan dan dukungan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Bapak Dr. Imam Mukhlash, S.Si, MT selaku Kepala Departemen Matematika ITS.
2. Ibu Prof. Dr. Erna Apriliani, M.Si selaku Dosen Wali yang telah memberikan arahan akademik selama penulis menempuh pendidikan di Departemen Matematika ITS.
3. Bapak Drs. Nurul Hidayat, M.Kom dan Bapak Drs. Soetrisno, M.Kom selaku dosen pembimbing Tugas Akhir yang telah banyak memberikan bimbingan, arahan serta motivasi sehingga Tugas Akhir ini dapat terselesaikan dengan baik.
4. Bapak Dr. Didik Khusnul Arif, S.Si, M.Si selaku Kepala Program Studi S1 Departemen Matematika ITS.
5. Bapak Drs. Iis Herisman, M.Si selaku Sekretaris Program Studi S1 Departemen Matematika ITS.
6. Ibu Dr. Dwi Ratna Sulistyaningrum, S.Si, MT dan Ibu Dra. Sri Suprapti H., M.Si selaku dosen penguji Tugas Akhir penulis.
7. Seluruh jajaran dosen dan staf Departemen Matematika ITS.

8. Keluarga tercinta yang senantiasa memberikan dukungan dan do'a.
9. Vina, Xenny, Metta, Faizah, Azaria selaku teman-teman satu bimbingan Tugas Akhir yang telah memberikan bantuan dan motivasi.
10. Sahabat-sahabat penulis, Airin, Afifah, Retno, Putri yang selalu memberikan semangat kepada penulis.
11. Neni, Nurma, Diana, Niken, Gery, Ivan, Wawan, Jessica, Eries, Mega, Lisa, Ayur dan Agus yang selalu menemani, memberikan motivasi dan membantu penulis.
12. Teman-teman angkatan 2013 yang saling mendukung dan memotivasi.
13. Semua pihak yang tak dapat penulis sebutkan satu-persatu, terima kasih telah membantu hingga terselesaikannya Tugas Akhir ini.

Penulis menyadari bahwa Tugas Akhir ini masih jauh dari kesempurnaan. Oleh karena itu, penulis mengharapkan kritik dan saran dari pembaca. Akhir kata, semoga Tugas Akhir ini dapat bermanfaat bagi semua pihak yang berkepentingan.

Surabaya, Juli 2017

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	Error! Bookmark not defined.
Abstrak	vi
<i>Abstract</i>	viii
KATA PENGANTAR.....	x
DAFTAR ISI	xii
DAFTAR GAMBAR.....	xvi
DAFTAR TABEL	xviii
DAFTAR LAMPIRAN	xx
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	3
1.5 Manfaat.....	4
1.6 Sistematika Penulisan Tugas Akhir	4
BAB II TINJAUAN PUSTAKA	5
2.1 Penelitian Terdahulu.....	5
2.2 Biometrika	7
2.3 Iris Mata.....	8
2.4 Sistem Pengenalan Iris Mata	9
2.5 Citra Digital	10
2.6 Segmentasi Citra.....	11
2.6.1 Deteksi Tepi Canny	11
2.6.2 Transformasi <i>Hough</i>	13
2.6.3 Daugman's Rubber Sheet Model.....	14
2.7 <i>Principal Component Analysis (PCA)</i>	15
2.8 <i>Learning Vector Quantization</i>	21
BAB III METODE PENELITIAN	25

3.1	Objek dan Aspek	25
3.2	Peralatan	25
3.3	Tahap Penelitian	26
3.4	Diagram Alir Penelitian.....	28
BAB IV PERANCANGAN DAN IMPLEMENTASI		
	PERANGKAT LUNAK	31
4.1	Perancangan Data	31
4.1.1	Data Masukan.....	31
4.1.2	Data Proses	32
4.1.3	Data Keluaran.....	33
4.2	Perancangan Proses Pengenalan iris Mata	33
4.2.1	Akuisisi Citra Mata	34
4.2.2	Lokalisasi Iris Mata	35
4.2.3	Normalisasi Iris Mata	40
4.2.4	Ekstraksi Ciri	42
4.2.5	Pencocokan Iris	42
4.3	Perancangan Antarmuka.....	45
4.4	Implementasi Perangkat Lunak	47
4.4.1	Implementasi Antarmuka	47
4.4.2	Implementasi Akuisisi Citra Iris Mata	50
4.4.3	Implementasi Lokalisasi Iris Mata	50
4.4.4	Implementasi Normalisasi Iris Mata	52
4.4.5	Implementasi Ekstraksi Ciri	54
4.4.6	Implementasi Pencocokan	55
BAB V UJI COBA DAN EVALUASI		57
5.1	Data Uji Coba.....	57
5.2	Uji Coba Proses Lokalisasi Iris Mata	57
5.2.1	Uji Coba Deteksi Tepi.....	57
5.2.2	Uji Coba Deteksi Batas Pupil dan Iris	60
5.2.3	Uji Coba Penutupan Bulu dan Kelopak Mata	60
5.3	Uji Coba Normalisasi	60

5.4	Uji Coba Ekstraksi Ciri.....	67
5.5	Uji Coba Pencocokan	67
5.6	Akurasi Sistem Pengenalan Iris	69
5.7	Penyebab Kegagalan Pengenalan Iris Mata.....	75
BAB VI PENUTUP.....		79
6.1	Kesimpulan.....	79
6.2	Saran	79
DAFTAR PUSTAKA.....		81
BIODATA PENULIS.....		93

DAFTAR GAMBAR

Gambar 2.1	Struktur permukaan iris mata	9
Gambar 2.2	Sistem pengenalan iris mata	10
Gambar 2.3	Pembagian arah gradien tepi	12
Gambar 2.4	Transformasi hough lingkaran.....	14
Gambar 2.5	Transformasi koordinat kartesian ke polar	14
Gambar 2.6	Contoh Arsitektur Jaringan LVQ.....	21
Gambar 3.1	Diagram alir tahap penelitian	28
Gambar 3.2	Diagram rancangan perangkat lunak	29
Gambar 4.1	Contoh citra mata database <i>CASIA</i> versi <i>1.0</i>	32
Gambar 4.2	Diagram blok proses pengenalan iris mata.....	34
Gambar 4.3	Diagram alir akuisisi citra mata.....	35
Gambar 4.4	Diagram alir proses lokalisasi iris mata.....	36
Gambar 4.5	Diagram alir proses deteksi tepi canny.....	37
Gambar 4.6	Diagram proses deteksi batas iris dan pupil	39
Gambar 4.7	Diagram proses penutupan bulu dan kelopak mata	40
Gambar 4.8	Diagram proses normalisasi citra iris mata	41
Gambar 4.9	Diagram alir ekstraksi ciri menggunakan PCA ...	43
Gambar 4.10	Diagram proses training LVQ	44
Gambar 4.11	Diagram alir proses testing LVQ.....	45
Gambar 4.12	Rancangan antarmuka halaman utama	46
Gambar 4.13	Rancangan antarmuka pengujian.....	47
Gambar 4.14	Antarmuka halaman utama.....	48
Gambar 4.15	Antarmuka halaman pengujian.....	49
Gambar 5.1	Hasil uji coba pencocokan.....	68
Gambar 5.2	Kegagalan deteksi batas iris karena tertutup oleh bulu mata	75
Gambar 5.3	Kegagalan pendeteksian batas luar iris.....	75
Gambar 5.4	Kondisi citra yang gagal dikenali karena perbedaan pendeteksian bulu dan kelopak mata	76

DAFTAR TABEL

Tabel 2.1	Perbandingan karakteristik biometrik	8
Tabel 4.1	Data proses identifikasi iris mata	33
Tabel 4.2	Kegunaan <i>tools</i> pada setiap panel	48
Tabel 5.1	Hasil uji coba deteksi tepi canny.....	58
Tabel 5.2	Lanjutan hasil uji coba deteksi tepi canny	59
Tabel 5.3	Hasil uji coba deteksi batas pupil dan iris.....	61
Tabel 5.4	Lanjutan hasil uji coba deteksi batas pupil dan iris.....	62
Tabel 5.5	Hasil uji coba penutupan bulu dan kelopak mata....	63
Tabel 5.6	Lanjutan hasil uji coba penutupan bulu dan kelopak mata.....	64
Tabel 5.7	Hasil uji coba normalisasi	65
Tabel 5.8	Lanjutan hasil uji coba normalisasi.....	66
Tabel 5.9	Contoh hasil vektor fitur dengan 10 vektor eigen ..	67
Tabel 5.10	Contoh hasil pengenalan metode LVQ	69
Tabel 5.11	Hasil akurasi uji coba 30 mata	74
Tabel 5.12	Hasil akurasi uji coba 50 mata	70
Tabel 5.13	Hasil akurasi uji coba 80 mata	70
Tabel 5.14	Hasil akurasi uji coba 108 mata	71
Tabel 5.15	Akurasi uji coba secara keseluruhan.....	73

DAFTAR LAMPIRAN

LAMPIRAN A	Hasil Uji Coba Pada 30 Mata Dengan 75 Vektor Eigen	83
LAMPIRAN B	Hasil Uji Coba Pada 50 Mata Dengan 75 Vektor Eigen	84
LAMPIRAN C	Hasil Uji Coba Pada 80 Mata Dengan 75 Vektor Eigen	86
LAMPIRAN D	Hasil Uji Coba Pada 108 Mata Dengan 75 Vektor Eigen	89

BAB I

PENDAHULUAN

Pada bab ini dijelaskan latar belakang, rumusan masalah, tujuan, manfaat dan sistematika penulisan dalam Tugas Akhir ini. Diharapkan dari penjelasan dalam bab ini gambaran secara umum dari Tugas Akhir dapat dipahami.

1.1 Latar Belakang

Pada era globalisasi ini, teknologi biometrik berkembang dengan cukup pesat. Hal ini disebabkan oleh kemajuan teknologi komputer dan tuntutan kebutuhan akan sistem keamanan yang canggih diberbagai bidang aplikasi. Teknologi biometrik merupakan teknologi yang memanfaatkan karakteristik biologis yang bersifat unik yang dimiliki oleh individu untuk proses identifikasi, seperti sidik jari, wajah, telapak tangan, suara dan iris mata. Sebelum adanya teknologi biometrik, sistem identifikasi yang umum digunakan adalah dengan cara konvensional seperti menggunakan *password* dan kunci [1]. Namun penggunaan teknologi konvensional ini memiliki beberapa kelemahan, seperti dapat hilang atau mudah dicuri, mudah terlupakan dan dapat diduplikasi. Oleh karena itu, teknologi biometrik dikembangkan untuk mengatasi kelemahan tersebut.

Sistem pengenalan individu dengan tingkat identifikasi yang tinggi, salah satunya dapat diperoleh dengan menggunakan teknologi biometrik berbasis iris mata. Hal ini dikarenakan iris mata memiliki tekstur yang unik dan cukup kompleks. Selain itu, dibandingkan dengan bagian tubuh manusia yang lain seperti sidikjari, iris mata memiliki pola yang lebih konsisten atau tidak berubah dalam waktu yang panjang [2]. Karena iris mata dilindungi oleh suatu lapisan transparan atau membran yang disebut dengan kornea, yang membuat polanya tidak mudah berubah oleh suatu kondisi ataupun benda asing. Iris mata memiliki karakteristik yang berbeda pada setiap individu seperti bintik-bintik, korona, garis-garis, crypts dan sebagainya [3]. Bahkan pada individu kembar identik sekalipun memiliki

karakteristik iris mata yang berbeda. Sehingga iris mata sangat efektif digunakan untuk identifikasi individu.

Dalam membangun sistem pengenalan berdasarkan iris mata, terdapat empat tahapan yang dapat dilakukan, yaitu: akuisisi citra, pra-pengolahan, ekstraksi ciri dan pencocokan. Sistem pengenalan individu menggunakan iris mata sudah banyak dikembangkan dengan berbagai metode. Salah satunya menggunakan metode *Principal Component Analysis* (PCA) yaitu suatu metode yang digunakan untuk ekstraksi ciri yang mampu mengidentifikasi ciri tertentu yang merupakan karakteristik suatu citra. Secara matematis PCA mentransformasikan sejumlah variabel yang berkorelasi menjadi beberapa variabel yang tidak berkorelasi tanpa menghilangkan informasi penting didalamnya [4]. Metode ini mampu mempercepat waktu komputasi dalam membangun sistem.

Pada pengenalan individu, proses pencocokan atau kalsifikasi diperlukan untuk mengelompokkan nilai-nilai piksel yang berdekatan kedalam kelas-kelas untuk mewakili beberapa fitur berdasarkan ciri dari citra individu tersebut. Metode *Learning Vector Quantization* (LVQ) merupakan salah satu metode klasifikasi yang dapat digunakan dalam pengenalan individu. Metode ini bertujuan untuk mendekati distribusi kelas vektor untuk meminimalkan kesalahan dalam pengklasifikasian. Dalam metode LVQ, lapisan kompetitif akan belajar secara otomatis untuk melakukan klasifikasi terhadap vektor-vektor input yang diberikan. Kelas-kelas yang didapat bergantung pada jarak antar vektor-vektor input. Jika vektor input mendekati sama maka lapisan kompetitif akan mengklasifikasikan kedua vektor input tersebut kedalam kelas yang sama.

Berdasarkan latar belakang tersebut, maka dalam Tugas Akhir ini dibangun sistem pengenalan individu melalui identifikasi iris mata menggunakan metode PCA untuk mendapatkan ciri dari citra dan LVQ untuk mengklasifikasikan citra tersebut.

1.2 Rumusan Masalah

Berdasarkan latar belakang, maka rumusan masalah yang digunakan dalam Tugas Akhir ini adalah sebagai berikut:

1. Melakukan pra-pengolahan terhadap setiap citra iris mata individu.
2. Mengaplikasikan metode PCA untuk mengekstraksi ciri citra iris mata pada setiap individu.
3. Menerapkan metode LVQ untuk mengenali individu pemilik iris mata.
4. Mengintegrasikan program pra-pengolahan, ekstraksi ciri dan pengenalan untuk membuat sistem perangkat lunak untuk mengenali individu melalui identifikasi iris mata.
5. Membuat *interface* sistem perangkat lunak pengenalan individu berbasis iris mata yang *user friendly*.

1.3 Batasan Masalah

Permasalahan dalam Tugas Akhir ini memiliki beberapa batasan sebagai berikut:

1. Database citra iris mata yang digunakan adalah data citra yang berasal dari *Chinese Academy of Science Institut of Automation (CASIA)*.
2. Data citra yang digunakan adalah data citra dengan format ekstensi *.bmp.
3. Pengujian dilakukan per citra iris mata.

1.4 Tujuan

Tujuan umum dalam usulan Tugas Akhir ini adalah untuk mendapatkan program yang dapat digunakan untuk mengenali individu (manusia). Sedangkan tujuan khusus dalam Tugas Akhir ini adalah sebagai berikut:

1. Mendapatkan program pra-pengolahan terhadap setiap citra iris mata individu.
2. Mendapatkan program untuk memunculkan ciri dari citra iris mata individu dengan mengaplikasikan metode PCA.
3. Mendapatkan program untuk mengidentifikasi iris mata individu dengan menggunakan LVQ.

1.5 Manfaat

Dari penulisan Tugas Akhir ini diharapkan memberikan manfaat sebagai berikut:

1. Dapat gunakan sebagai langkah awal pembuatan sistem keamanan yang berbasis pengenalan iris mata.
2. Sebagai referensi bagi pembaca dalam penelitian selanjutnya.

1.6 Sistematika Penulisan Tugas Akhir

Sistematika penulisan dalam Tugas Akhir ini adalah sebagai berikut:

1. BAB I PENDAHULUAN

Bab ini berisi tentang gambaran umum dari Tugas Akhir yang meliputi latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan.

2. BAB II TINJAUAN PUSTAKA

Bab ini berisi tentang teori dasar yang berhubungan dengan pokok pembahasan yang mendukung dalam Tugas Akhir.

3. BAB III METODOLOGI PENELITIAN

Bab ini membahas objek dan aspek dalam Tugas Akhir, peralatan, serta langkah-langkah yang digunakan dalam penyusunan Tugas Akhir.

4. BAB IV PERANCANGAN DAN IMPLEMENTASI PERANGKAT LUNAK

Bab ini berisi tentang perancangan proses, algoritma sistem yang akan dibangun dan kemudian dilanjutkan dengan pengimplementasian kedalam bahasa pemrograman agar dapat menghasilkan perangkat lunak yang dapat digunakan untuk pengenalan iris mata.

5. BAB V UJI COBA DAN EVALUASI SISTEM

Bab ini membahas tentang pelaksanaan percobaan dan hasil dari sistem yang telah dibangun dan kemudian mengevaluasi uji coba tersebut.

6. BAB VI KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan Tugas Akhir dari bab uji coba dan evaluasi serta saran untuk penelitian lebih lanjut.

BAB II

TINJAUAN PUSTAKA

Pada bab ini diuraikan mengenai dasar teori yang digunakan dalam penyusunan Tugas Akhir ini. Dasar teori yang dijelaskan dibagi menjadi beberapa sub bab yaitu Penelitian Terdahulu, Biometrika, Sistem Pengenalan Iris Mata, Citra Digital, Deteksi Tepi Canny, Transformasi *Hough*, *Daugman's Rubber Sheet Model*, *Principal Component Analysis* dan *Learning Vector Quantization*.

2.1 Penelitian Terdahulu

Penelitian mengenai pengenalan iris mata cukup banyak mendapat perhatian dan cukup berkembang. Pengenalan iris mata dapat diaplikasikan dalam berbagai bidang misalnya pada sistem keamanan, pencarian identitas individu dan lainnya.

Salah satu penelitian tentang pengenalan iris mata adalah penelitian yang dilakukan oleh M. Rizky Faundra pada tahun 2011 dengan penelitian yang berjudul “Aplikasi *Filter Log Gabor* Pada Sistem Pengenalan Iris Mata”. Pada penelitian ini metode *filter log gabor* digunakan untuk mengekstraksi ciri dari citra iris mata tersebut. Sedangkan pada proses pencocokan, metode yang digunakan adalah *Modified Hamming Distance* (Jarak Hamming Termodifikasi) yang mana proses ini mencocokkan kesamaan 2 citra iris mata. Dalam memutuskan cocok atau tidaknya iris mata tersebut, Rizky menggunakan nilai ambang sebesar 0.4 yang artinya jika jarak *Hamming* kurang dari sama dengan 0.4 maka iris mata diputuskan cocok. Hasil akurasi dari sistem pengenalan iris mata dengan metode *filter log gabor* sebesar 94% [5]. Pada penelitian ini database citra mata yang digunakan adalah database citra mata *CASIA versi 1.0* dari 30 individu.

Penelitian yang membahas tentang ekstraksi ciri menggunakan metode PCA dilakukan oleh Nor'aini A.J., Rohilah S., Azilah S. pada tahun 2013 dalam penelitiannya yang berjudul “*Clasification of Iris Recognition using Principal Component Analysis dan Support Vector Machine*”. Pada

penelitian ini metode PCA dan *Support Vector Machine-Radial Basis Function* (SVM-RBF) digunakan untuk mengklasifikasikan vagina dan panggul dari daerah iris. Akurasi klasifikasi tertinggi diperoleh sebesar 75% untuk vagina dan panggul dengan menggunakan kedua pendekatan tersebut, SVM-RBF dan SVM-RBF dengan PCA. Namun, akurasi klasifikasi tertinggi dari vagina diperoleh sebesar 80% dengan menggunakan SVM-RBF dengan PCA sementara panggul mencapai 70% dari persentase rata-rata matriks *confusion* [6].

Penelitian lain yang membahas mengenai pengenalan iris mata menggunakan metode PCA adalah penelitian yang berjudul “*Iris Recognition based on PCA for Person Identification*” oleh Aniket S.Buddharpawar dan S.Subbaraman yang dilakukan pada tahun 2015. Dalam penelitiannya data citra yang digunakan berasal dari 36 individu dengan 7 citra pada setiap individu, yang mana database citra mata diambil dari database *CASIA versi 1.0*. Metode PCA diterapkan untuk mendapatkan ciri citra iris mata dan jarak *euclidean* digunakan untuk proses klasifikasi. Pada tahap *preprocessing* Aniket menambahkan *image enhancement* menggunakan *histrogram equalization* untuk memperoleh fitur yang lebih akurat. Hasil akurasi dari penelitian ini diperoleh sebesar 85% [7].

Penelitian yang membahas tentang klasifikasi dalam pengenalan iris mata salah satunya adalah penelitian yang berjudul “Identifikasi Iris Mata Menggunakan Tapis *Gabor Wavelet* dan Jaringan Syaraf Tiruan *Learning Vector Quantization* (LVQ)” oleh R. Rizal Isnanto pada tahun 2009. Dalam penelitian tersebut metode *Tapir Gabor Wavelet* untuk mengekstraksi ciri dari citra yang ternormalisasi dan LVQ digunakan untuk klasifikasi selain itu pengenalan dengan jarak *Euclidean* digunakan untuk pembandingan dalam pengenalan citra iris. Persentase pengenalan terbaik adalah sebesar 87,5% [8]. Akan tetapi pada penelitian ini, dalam pengambilan citra, data citra iris mata yang digunakan kurang baik sehingga mempengaruhi proses pengenalan.

2.2 Biometrika

Biometrika merupakan sebuah sistem pengenalan pola yang didasarkan pada keunikan yang ada pada masing-masing individu. Teknologi biometrika digunakan untuk menunjukkan keaslian (autentikasi) dari individu. Autentikasi adalah konsep yang menunjukkan bahwa hanya mereka yang diijinkan yang mempunyai akses terhadap hal-hal yang bersifat rahasia lainnya.







Tidak semua bagian tubuh atau perilaku seseorang dapat digunakan sebagai biometrika. Ada beberapa persyaratan yang harus dipenuhi agar bagian-bagian tubuh atau perilaku manusia dapat digunakan sebagai biometrika, antara lain [1]:

1. Universal (*universality*), artinya karakteristik yang dipilih harus dimiliki oleh setiap orang.
2. Membedakan (*distinctiveness*), artinya karakteristik yang dipilih memiliki kemampuan membedakan antara satu orang dengan orang lain.
3. Permanen (*permanence*), artinya karakteristik yang dipilih tidak cepat berubah dalam periode waktu yang lama.
4. Kolektabilitas (*collectability*), artinya karakteristik yang dipilih mudah diperoleh dan dapat diukur secara kuantitatif.
5. Unjuk kerja (*performance*), artinya karakteristik yang dipilih dapat memberikan unjuk kerja yang bagus baik dari segi akurasi maupun kecepatan, termasuk sumber daya yang dibutuhkan untuk memperolehnya.
6. Dapat diterima (*acceptability*), artinya masyarakat mau menerima karakteristik yang digunakan.
7. Tidak mudah dikelabui (*circumvention*), artinya karakteristik yang dipilih tidak mudah dikelabui dengan berbagai cara curang.

Syarat 1–4 merupakan syarat utama yang harus dipenuhi, sedangkan 5-7 merupakan syarat tambahan. Bila karakteristik yang dipilih memenuhi persyaratan di atas maka karakteristik tersebut dapat digunakan untuk biometrika.

Perbandingan karakteristik biometrika menurut Anil [9] berdasarkan syarat utama biometrika dapat dilihat pada Tabel 2.1:

Tabel 2.1 Perbandingan karakteristik biometrik

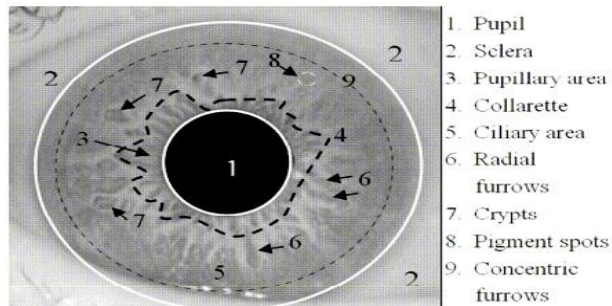
					
Biometrik	Sidik Jari	Wajah	Telapak Tangan	Iris Mata	Suara
Penghalang Universal	Gangguan jari	Tidak ada	Gangguan tangan	Tuna-netra	Tuna wicara
Membedakan	Tinggi	Rendah	Sedang	Tinggi	Rendah
Permanen	Tinggi	Sedang	Sedang	Tinggi	Rendah
Kolektabilitas	Sedang	Tinggi	Tinggi	Sedang	Sedang
Performasi	Tinggi	Rendah	Sedang	Tinggi	Rendah
Dapat diterima	Sedang	Tinggi	Sedang	Rendah	Tinggi
Dapat dikelabui	Sedang	Tinggi	Sedang	Rendah	Tinggi

Dari tabel perbandingan diatas dapat diketahui bahwa iris mata memiliki rata-rata karakteristik yang tinggi dibandingkan dengan karakteristik yang lainnya, sehingga iris mata sangat efektif digunakan pada pengenalan individu.

2.3 Iris Mata

Iris mata adalah daerah berbentuk gelang pada mata yang dibatasi oleh pupil dan *sclera* (bagian putih dari mata) [1]. Pewarnaan dan struktur iris terhubung secara genetik sedangkan rincian polanya tidak. Tekstur dari iris mata dibentuk selama perkembangan janin dan menstabilkan diri sepanjang dua tahun pertama kehidupan janin. Tekstur iris mata yang kompleks membawa informasi yang sangat unik dan bermanfaat untuk pengenalan individu. Meskipun memiliki genetik yang serupa, seseorang akan tetap memiliki pola iris yang berbeda. Bahkan pola iris mata pada satu individu pun memiliki pola yang berbeda pada bagian mata kiri dan kanan.

Pengenalan iris mata adalah suatu proses untuk mengenali seseorang dengan menganalisa pola acak dari iris mata. Dimana pengenalan ini bertujuan untuk mengenali objek dengan cara mengekstraksi informasi yang terdapat dalam citra mata tersebut. Pola acak pada iris mata yang membedakan adalah pola yang ada pada daerah *zigzag collarette*. Struktur permukaan mata yang menunjukkan iris mata dapat dilihat pada Gambar 2.1.



Gambar 2.1 Struktur permukaan iris mata

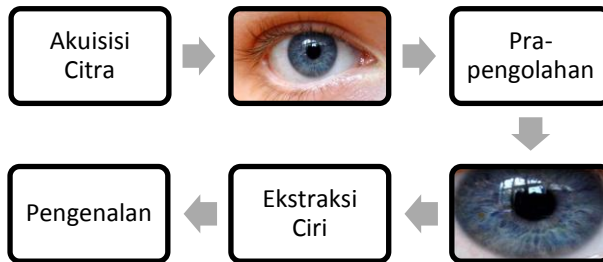
Kelebihan dari pengenalan dengan iris mata dibandingkan dengan bagian tubuh manusia yang lain seperti sidik jari, iris mata memiliki pola/tekstur yang lebih konsisten atau tidak berubah dalam waktu yang panjang [2]. Akan tetapi, pada tahap akuisisi data, alat yang digunakan relatif mahal karena alat akuisis harus menjamin kenyamanan pengguna dalam memakainya [1].

2.4 Sistem Pengenalan Iris Mata

Dalam membangun sistem pengenalan iris mata, maka dilakukan tahapan yang terdiri atas akuisisi citra, pra-pengolahan, ekstraksi ciri dan pengenalan. Tahapan dalam pengenalan iris mata dapat dilihat pada Gambar 2.2.

- a. Akuisisi citra: tahap awal untuk mendapatkan citra digital. Tujuan akuisisi citra adalah untuk mendapatkan citra iris dari individu yang digunakan untuk pendaftaran maupun pengujian.

- b. Pra-pengolahan: proses pembuatan iris mata yang telah diakuisisi menjadi file citra ternormalisir yang siap untuk diekstraksi cirinya.
- c. Ekstraksi ciri: proses untuk mendapatkan ciri dari citra iris mata.
- d. Pengenalan: proses pengukuran kesamaan antara nilai dari citra input dengan nilai yang berada dalam database.



Gambar 2.2 Sistem pengenalan iris mata

2.5 Citra Digital

Citra digital merupakan fungsi dua dimensi yang dapat dinyatakan dengan fungsi $f(x,y)$ yang berukuran M baris dan N kolom, dimana x dan y merupakan titik koordinat spasial. Sedangkan amplitudo dari fungsi f pada sembarang koordinat (x,y) merupakan nilai intensitas atau tingkat keabuan dari citra pada titik tersebut [10]. Apabila nilai x,y dan nilai amplitudo f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital. Citra digital dapat direpresentasikan dalam bentuk matriks sebagai berikut:

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & \ddots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

2.6 Segmentasi Citra

Segmentasi merupakan teknik untuk membagi suatu citra menjadi beberapa daerah (*region*) dimana setiap daerah memiliki kemiripan atribut [10]. Tujuan dari segmentasi adalah untuk menyederhanakan atau mengubah penyajian gambar ke sesuatu yang lebih bermakna dan lebih mudah untuk dianalisis. Gambar segmentasi biasanya digunakan untuk menemukan obyek dan batas-batas (garis, kurva, dll) dalam gambar. Segmentasi citra merupakan tahapan yang sangat penting dalam proses pengenalan pola sebab hasil dari citra yang tersegmentasi tersebut akan digunakan dalam proses selanjutnya, seperti proses ekstraksi ciri atau proses klasifikasi. Beberapa tahapan yang digunakan dalam segmentasi citra pada Tugas Akhir ini diantaranya adalah deteksi tepi canny, transformasi *Hough* dan *Daugman's Rubber Sheet Model*.

2.6.1 Deteksi Tepi Canny

Deteksi tepi berfungsi untuk mengidentifikasi garis batas (*boundary*) dari suatu obyek yang terdapat pada citra. Deteksi tepi umumnya merupakan langkah awal melakukan segmentasi citra. Salah satu algoritma deteksi tepi modern adalah deteksi tepi menggunakan metode *canny*.

Operator Canny, yang dikemukakan oleh John Canny pada tahun 1986, terkenal sebagai operator deteksi tepi yang optimal. Algoritma ini memberikan tingkat kesalahan yang rendah dalam melokalisasi titik-titik tepi (jarak antara piksel-piksel tepi yang ditemukan pada deteksi dan tepi yang sesungguhnya sangat pendek), dan hanya memberikan satu tanggapan untuk satu tepi. Algoritma deteksi tepi canny dapat dijelaskan sebagai berikut [1]:

1. Mengurangi noise pada citra menggunakan filter *gaussian*.

Persamaan 2.1 merupakan perhitungan filter *gaussian* pada citra yang menggunakan persamaan *gaussian* 2D.

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (2.1)$$

dimana σ (sigma) menyatakan standart deviasi, x dan y merupakan koordinat titik.

2. Mencari nilai tepi dengan menghitung gradien citra tersebut. Perhitungan gradien ini dapat dilakukan melalui persamaan 2.2:

$$|G| = |G_x| + |G_y| \quad (2.2)$$

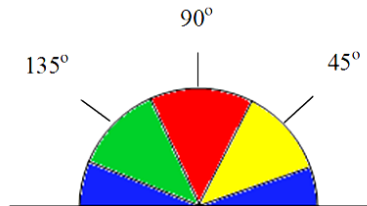
dimana G merupakan intensitas gradien, G_x gradien arah sumbu x , G_y gradien arah sumbu y

3. Langkah selanjutnya adalah mencari arah tepi dari nilai tepi yang telah ditemukan dengan persamaan 2.3:

$$\theta = \tan^{-1} \frac{G_y}{G_x} \quad (2.3)$$

dimana θ adalah arah gradien.

4. Menghubungkan arah tepi ke arah yang dapat ditelusuri sesuai dengan gambar aslinya. Kemudian dibagi kedalam empat warna sehingga garis dengan arah berbeda dan memiliki warna yang berbeda. Berikut adalah pembagian arah yang berlaku:



Gambar 2.3 Pembagian arah gradien tepi

- Semua arah tepi yang berkisar antara 0 dan 22,5 serta 157,5 dan 180 derajat (warna biru) diubah menjadi 0 derajat.
- Semua arah tepi yang berkisar antara 22,5 dan 67,5 derajat (warna kuning) diubah menjadi 45 derajat.
- Semua arah tepi yang berkisar antara 67,5 dan 112,5 derajat (warna merah) diubah menjadi 90 derajat.
- Semua arah tepi yang berkisar antara 112,5 dan 157,5 derajat (warna hijau) diubah menjadi 135 derajat.

5. Setelah arah tepi diketahui, proses *non-maximum suppression*. *Non-maximum suppression* digunakan untuk menelusuri tepi yang ada didalam arah tepi dan meredam nilai pixel manapun (dijadikan 0) yang tidak dianggap layak untuk menjadi sebuah tepi. Proses ini nantinya akan menampilkan garis tipis pada output citra.
6. Langkah terakhir adalah proses *hysteresis* yang digunakan untuk menghilangkan garis-garis yang seperti terputus-putus pada tepi obyek. Caranya adalah dengan menggunakan dua ambang T1 dan T2. Lalu, semua piksel citra yang bernilai lebih besar daripada T1 dianggap sebagai piksel tepi. Selanjutnya, semua piksel yang terhubung dengan piksel tersebut dan memiliki nilai lebih besar dari T2 juga dianggap sebagai piksel tepi.

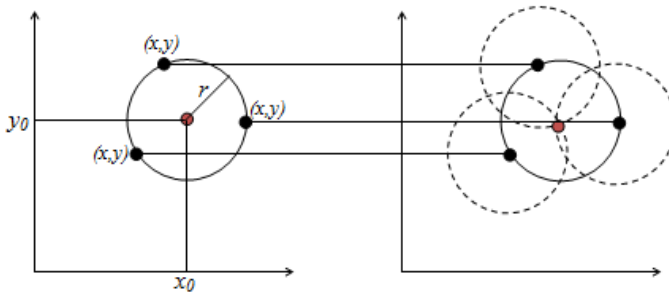
2.6.2 Transformasi Hough

Hough Transform pertama kali diperkenalkan oleh Paul Hough pada tahun 1962 untuk mendeteksi garis lurus. *Hough Transform* adalah teknik transformasi citra yang dapat digunakan untuk mengisolasi suatu objek pada citra dengan menemukan batas-batasnya (*boundary detection*). Karena tujuan dari sebuah transformasi adalah untuk mendapatkan suatu fitur yang lebih spesifik, *classical hough transform* merupakan teknik yang paling umum digunakan untuk mendeteksi objek yang berbentuk kurva seperti garis, lingkaran, elips, dan parabola. Dalam subbab ini hanya dibahas transformasi *hough* lingkaran karena salah satu tahap utama dalam pengenalan iris mata adalah deteksi lingkaran pada pupil dan iris mata. Prosedur yang digunakan dalam mendeteksi lingkaran adalah sama dengan transformasi *hough* pada garis akan tetapi dikerjakan pada ruang 3D (x_0, y_0, r) . Untuk mencari titik pusat dan jari-jari pada lingkaran iris dan pupil dapat diperoleh dengan persamaan 2.4 berikut [10]:

$$(x - x_0)^2 + (y - y_0)^2 = r^2 \quad (2.4)$$

dimana (x, y) koordinat titik-titik lingkaran, (x_0, y_0) koordinat pusat lingkaran dan r merupakan jari-jari lingkaran.

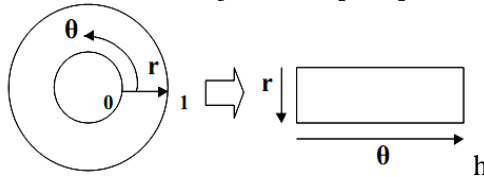
Pada proses pendeteksian batas iris dan pupil, transformasi *hough* lingkaran digunakan untuk mencari daerah iris dengan cara mencari radius dan titik pusat lingkaran pupil dan iris. Radius dan titik pusat dari pupil dan iris didapatkan dengan cara menggambar lingkaran buatan pada setiap titik-titik tepi yang ditemukan dengan jari-jari sebesar r . Kemudian dilakukan voting pada titik perpotongan lingkaran buatan. Titik koordinat yang paling banyak dilewati oleh lingkaran buatan menjadi titik koordinat pusat lingkaran dan jari-jari lingkaran buatan menjadi jari-jari lingkaran. Gambar 2.4 mengilustrasikan transformasi *hough* dalam koordinat (x_0, y_0, r) .



Gambar 2.4 Transformasi *hough* lingkaran

2.6.3 Daugman's Rubber Sheet Model

Untuk normalisasi citra iris ke bidang persegi, *homogenous rubber sheet model* digunakan oleh Daugman. Gambar 2.5 menunjukkan gambaran model dari *Daugman Rubber Sheet Model* yang mana model ini meletakkan tiap titik dalam daerah iris ke sebuah pasangan koordinat polar (r, θ) dimana r berada di interval $[0, 1]$ dan θ adalah derajat antara $[0, 2\pi]$.



Gambar 2.5 Transformasi koordinat kartesian ke polar

Proses normalisasi *daugman rubber sheet model* dimulai dari perhitungan jarak antara tepi pupil dan iris melalui persamaan 2.5 berikut [11]:

$$r = \sqrt{\alpha\beta} \pm \sqrt{\alpha\beta^2 - \alpha - r_1^2} \quad (2.5)$$

dengan

$$\alpha = O_x^2 + O_y^2$$

$$\beta = \cos^2\left(\pi - \arctan\left(\frac{O_y}{O_x}\right) - \theta\right)$$

dimana r adalah jarak antara tepi pupil dan iris pada sudut tertentu. O_x dan O_y merupakan jarak dari pusat pupil relatif terhadap pusat iris. Sedangkan θ dan r_1 secara terurut adalah sudut kemiringan dan jari-jari dari iris.

Tahap selanjutnya adalah menghitung koordinat titik-titik data pada lingkaran iris yang kemudian dilakukan pemetaan titik-titik dari koordinat kartesian ke koordinat polar. Proses pemetaan dimodelkan sebagai [11]:

$$I(x(r, \theta), y(r, \theta)) \rightarrow I(r, \theta) \quad (2.6)$$

dengan $x(r, \theta) = (1 - r)x_p(\theta) + rx_i(\theta)$

$$y(r, \theta) = (1 - r)y_p(\theta) + ry_i(\theta)$$

dimana $I(x, y)$ adalah daerah iris, (x, y) koordinat cartesian asal, (r, θ) koordinat polar yang telah dinormalisasi, dan x_p, y_p dan x_i, y_i adalah koordinat pupil dan batas iris pada arah θ .

Rubber Sheet Model mempertimbangkan dilasi pupil dan inkonsistensi ukuran untuk menghasilkan sebuah representasi normalisasi dengan dimensi tetap. Dengan demikian, daerah iris mata dimodelkan sebagai sebuah *flexible rubber sheet* berdasarkan pada batas iris dengan pusat pupil sebagai titik acuan.

2.7 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) merupakan salah satu metode yang digunakan untuk mereduksi fitur atau dimensi. Secara matematis PCA mentransformasikan sejumlah variabel yang berkorelasi menjadi beberapa variabel yang tidak

berkolerasi tanpa menghilangkan informasi penting didalamnya [2]. PCA menghasilkan sejumlah vektor basis orthonormal dalam bentuk kumpulan vektor eigen dari suatu matriks kovarian tertentu, yang dapat secara optimal merepresentasikan distribusi data. Kumpulan vektor basis tersebut digunakan untuk membentuk suatu sub ruang data dengan dimensi yang lebih kecil dan merepresentasikan data baru dari data awal. Vektor-vektor basis yang membentuk ruang baru diperoleh dari proses pencarian vektor eigen (*eigen vector*) dan nilai eigen (*eigen value*) dari suatu matriks kovarian tertentu, sehingga ruang baru tersebut dapat disebut dengan ruang *eigen*. Secara singkat langkah-langkah dari proses pembentukan ruang eigen menggunakan PCA adalah sebagai berikut [12]:

Misalkan terdapat sejumlah n citra dan setiap citra berukuran $M \times N$. Kemudian setiap citra yang berukuran $M \times N$ direpresentasikan kedalam vektor kolom sehingga setiap citra berukuran $MN \times 1$. Kemudian didefinisikan sebuah matriks X merupakan kumpulan dari vektor-vektor kolom yang mewakili setiap citra. Sehingga X dapat diperoleh:

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \ddots & x_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{MN,1} & x_{MN,2} & \cdots & x_{MN,n} \end{bmatrix} \quad (2.7)$$

Dimana $X_{j,1}$ merupakan vektor kolom dari citra pertama dengan $j = 1, 2, \dots, MN$ dan MN adalah total piksel dari citra. Kemudian menghitung rata-rata citra dengan rumus:

$$\bar{x}_j = \frac{1}{n} \sum_{i=1}^n x_{j,i} \quad (2.8)$$

Dimana n adalah banyaknya citra dan \bar{x}_j adalah rata-rata dari baris ke- j pada matriks X dengan $j = 1, 2, \dots, MN$. Maka matriks normalisasi diperoleh sebagai berikut:

$$A = \begin{bmatrix} x_{1,1} - \bar{x}_1 & x_{1,2} - \bar{x}_1 & \cdots & x_{1,n} - \bar{x}_1 \\ x_{2,1} - \bar{x}_2 & x_{2,2} - \bar{x}_2 & \ddots & x_{2,n} - \bar{x}_2 \\ \vdots & \vdots & & \vdots \\ x_{MN,1} - \bar{x}_{MN} & x_{MN,2} - \bar{x}_{MN} & \cdots & x_{MN,n} - \bar{x}_{MN} \end{bmatrix} \quad (2.9)$$

Matriks A adalah matriks normalisasi yang dapat digunakan untuk menghitung matriks kovarian dengan persamaan:

$$C = \frac{1}{n-1} AA^T \quad (2.10)$$

Dimana C adalah matriks kovarian dan A^T adalah transpose matriks normalisasi. Kemudian dari hasil kovarian matriks dicari nilai eigen dan vektor eigen dengan persamaan:

$$Cv = \lambda v \quad (2.11)$$

dengan λ adalah nilai eigen dan v adalah vektor eigen. Nilai eigen yang didapat kemudian diurutkan dari nilai terbesar sampai nilai terkecil dan vektor eigen disusun sesuai dengan nilai eigen yang telah diurutkan. Jika $[\lambda_1 \lambda_2 \lambda_3 \dots \lambda_n]$ dan $[v_1 v_2 v_3 \dots v_n]$ adalah PCA atau vektor eigen dan nilai eigen dari matriks kovarians C yang terpilih. Maka diperoleh vektor fitur dengan rumus:

$$w = v^T A \quad (2.12)$$

dimana v merupakan vektor eigen terpilih dan w merupakan vektor fitur.

Contoh perhitungan PCA adalah sebagai berikut.

Misalkan diberikan tiga buah matriks yang dianggap sebagai piksel dari citra dengan ukuran 3×3 , yaitu:

$$x_1 = \begin{bmatrix} 2 & 3 & 4 \\ 1 & 3 & 7 \\ 6 & 4 & 9 \end{bmatrix}; x_2 = \begin{bmatrix} 5 & 3 & 7 \\ 1 & 1 & 7 \\ 0 & 4 & 2 \end{bmatrix}; \text{ dan } x_3 = \begin{bmatrix} 2 & 3 & 4 \\ 1 & 4 & 5 \\ 4 & 4 & 7 \end{bmatrix}$$

Kemudian masing-masing matriks diubah menjadi vektor kolom yang digabungkan menjadi satu kesatuan matriks sehingga terbentuk matriks baru.

$$X = [x_1 \ x_2 \ x_3]$$

$$X = \begin{bmatrix} 2 & 5 & 2 \\ 3 & 3 & 3 \\ 4 & 7 & 4 \\ 1 & 1 & 1 \\ 3 & 1 & 4 \\ 7 & 7 & 5 \\ 6 & 0 & 4 \\ 4 & 4 & 4 \\ 9 & 2 & 7 \end{bmatrix}$$

Selanjutnya matriks baru yang terbentuk dicari nilai rata-rata pada tiap barisnya, hasil rata-rata \bar{x} tiap baris $j=1,2,...,9$ dari matriks X adalah

$$\bar{x}_j = \frac{[x_{1j} + x_{2j} + x_{3j}]}{n} = \begin{bmatrix} \bar{x}_1 \\ \bar{x}_2 \\ \bar{x}_3 \\ \bar{x}_4 \\ \bar{x}_5 \\ \bar{x}_6 \\ \bar{x}_7 \\ \bar{x}_8 \\ \bar{x}_9 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \\ 5 \\ 1 \\ 2.6667 \\ 6.3333 \\ 3.3333 \\ 4 \\ 6 \end{bmatrix}$$

Setelah diketahui nilai rata-ratanya, langkah selanjutnya adalah mencari matriks normalisasi dengan cara menghitung selisih tiap matriks kolom dengan rata-rata tiap baris.

$$A = X - \bar{x}_j = [x_{1j} - \bar{x}_j \ x_{2j} - \bar{x}_j \ x_{3j} - \bar{x}_j]$$

$$A = \begin{bmatrix} 2-3 & 5-3 & 2-3 \\ 3-3 & 3-3 & 3-3 \\ 4-5 & 7-5 & 4-5 \\ 1-1 & 1-1 & 1-1 \\ 3-2.6667 & 1-2.6667 & 4-2.6667 \\ 7-6.3333 & 7-6.3333 & 5-6.3333 \\ 6-3.3333 & 0-3.3333 & 4-3.3333 \\ 4-4 & 4-4 & 4-4 \\ 9-6 & 2-6 & 7-6 \end{bmatrix}$$

$$= \begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \\ 0.3333 & -1.6667 & 1.3333 \\ 0.6667 & 0.6667 & -1.3333 \\ 2.6667 & -3.3333 & 0.6667 \\ 0 & 0 & 0 \\ 3 & -4 & 1 \end{bmatrix}$$

Kemudian menghitung matriks kovarian dengan mengalikan matriks normalisasi dengan transposenya.

$$C = \frac{1}{n-1} AA^T$$

$$C = \frac{1}{3-1} \begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \\ 0.3333 & -1.6667 & 1.3333 \\ 0.6667 & 0.6667 & -1.3333 \\ 2.6667 & -3.3333 & 0.6667 \\ 0 & 0 & 0 \\ 3 & -4 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & -1 & 0 & 0.3333 & 0.6667 & 2.6667 & 0 & 3 \\ 2 & 0 & 2 & 0 & -1.6667 & 0.6667 & -3.3333 & 0 & -4 \\ -1 & 0 & -1 & 0 & 1.3333 & -1.3333 & 0.6667 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 3 & 0 & 3 & 0 & -2.5 & 1 & -5 & 0 & -6 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 3 & 0 & -2.5 & 1 & -5 & 0 & -6 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -2.5 & 0 & -2.5 & 0 & 2.3333 & -1.3333 & 3.6667 & 0 & 4.5 \\ 1 & 0 & 1 & 0 & -1.3333 & 1.3333 & -0.6667 & 0 & -1 \\ -5 & 0 & -5 & 0 & 3.6667 & -0.6667 & 9.3333 & 0 & 11 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -6 & 0 & -6 & 0 & 4.5 & 1 & 11 & 0 & 13 \end{bmatrix}$$

Dari matriks kovarian tersebut kemudian dicari nilai eigen dan vektor eigennya, diperoleh

$$V = \begin{bmatrix} -0.0294 & 0.3031 & 0 & 0.7071 & 0 & 0.4981 & 0.1249 & 0.2148 & -0.3121 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.0294 & 0.3031 & 0 & -0.7071 & 0 & 0.4981 & 0.1249 & 0.2148 & -0.3121 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -0.0160 & 0.8191 & 0 & 0 & 0 & -0.1232 & -0.0267 & -0.5039 & 0.2429 \\ -0.0932 & 0.3748 & 0 & 0 & 0 & -0.5516 & -0.1464 & 0.7213 & -0.0696 \\ 0.7408 & 0.0630 & 0 & 0 & 0 & 0.1949 & 0.1287 & 0.2917 & 0.5546 \\ 0.1029 & 0.0072 & 0 & 0 & 0 & 0.2435 & -0.9644 & 0 & 0 \\ -0.6557 & -0.0282 & 0 & 0 & 0 & 0.2951 & 0.0043 & 0.2201 & 0.6586 \end{bmatrix}$$

dan $D = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 2.2857 \ 29.7143]$

Vektor eigen kemudian diurutkan berdasarkan urutan nilai eigen dari yang terbesar ke terkecil dan dipilih sejumlah nilai eigen v sebagai *Principal Component* (PC). Misalkan dipilih 2 nilai eigen sebagai PC sehingga diperoleh

$$V = v = \begin{bmatrix} -0.3121 & 0.2148 \\ 0 & 0 \\ -0.3121 & 0.2148 \\ 0 & 0 \\ 0.2429 & -0.5039 \\ -0.0696 & 0.7213 \\ 0.5546 & 0.2917 \\ 0 & 0 \\ 0.6586 & 0.2201 \end{bmatrix}$$

Proses terakhir dari algoritma ini adalah mencari nilai fitur vektor dengan mengalikan tranpose dari vektor eigen dengan matriks normalisasi, diperoleh

$$w = v^T A$$

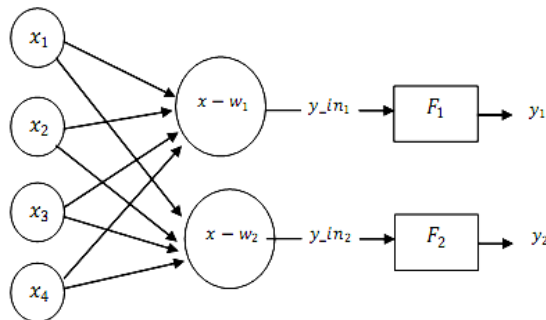
$$w = \begin{bmatrix} -0.3121 & 0 & -0.3121 & 0 & 0.2429 & -0.0696 & 0.5546 & 0 & 0.6586 \\ 0.2148 & 0 & 0.2148 & 0 & -0.5039 & 0.7213 & 0.2917 & 0 & 0.2201 \end{bmatrix} \begin{bmatrix} -1 & 2 & -1 \\ 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \\ 0.3333 & -1.6667 & 1.3333 \\ 0.6667 & 0.6667 & -1.3333 \\ 2.6667 & -3.3333 & 0.6667 \\ 0 & 0 & 0 \\ 3 & -4 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 4.1135 & -6.1827 & 2.0692 \\ 1.3215 & 0.3272 & -1.6486 \end{bmatrix}$$

Matriks w merupakan vektor fitur yang mana pada tiap kolom mewakili fitur pada setiap citra.

2.8 *Learning Vector Quantization*

Learning Vector Quantization adalah suatu metode untuk melakukan pembelajaran pada lapisan-lapisan kompetitif yang terwarisi. Suatu lapisan kompetitif akan belajar secara otomatis untuk melakukan klasifikasi terhadap vektor-vektor input yang diberikan. Gambar 2.6 merupakan contoh arsitektur jaringan LVQ yang memiliki empat neuron pada lapisan masukan dan dua neuron pada lapisan keluaran. Proses yang terjadi pada setiap neuron adalah mencari jarak antara vektor masukan ke bobot yang bersangkutan (w_1 dan w_2). w_1 adalah vektor bobot yang menghubungkan setiap neuron pada lapisan masukan ke neuron pertama pada lapisan keluaran, sedangkan w_2 adalah vektor bobot yang menghubungkan setiap neuron pada lapisan masukan ke neuron kedua pada lapisan keluaran. Fungsi aktivasi F_1 akan memetakan y_{in1} ke $y_1=1$ apabila $|x - w_1| < |x - w_2|$ dan $y_1=0$ jika sebaliknya. Demikian juga Fungsi aktivasi F_2 akan memetakan y_{in2} ke $y_2=1$ apabila $|x - w_2| < |x - w_1|$ dan $y_2=0$ jika sebaliknya.



Gambar 2.6 Contoh Arsitektur Jaringan LVQ

Algoritma dari pembelajaran LVQ secara umum adalah sebagai berikut [13]:

Step 0 : Inisialisasi vektor referensi
 Inisialisasi learning rate α , $\alpha(0)$.

Step 1 : bila kondisi STOP belum
 terpenuhi, kerjakan Step 2-6

Step 2 : Untuk setiap vektor training x , kerjakan
 Step 3-4

Step 3 : Dapatkan j sedemikian hingga $|x - w_j|$
 minimum

Step 4 : Update W_j sebagai berikut:
 Jika $T = C_j$ maka
 $W_j(\text{baru}) = W_j(\text{lama}) + \alpha(x - W_j(\text{lama}))$
 Jika $T \neq C_j$ maka
 $W_j(\text{baru}) = W_j(\text{lama}) - \alpha(x - W_j(\text{lama}))$

Step 5 : Reduksi learning rate (α)

Step 6 : Tes kondisi STOP
 Uji kondisi stop ini dapat dilakukan
 berdasarkan jumlah iterasi tertentu
 (mulai dari langkah 1) atau setelah laju
 pembelajaran yang telah mencapai harga
 yang sangat kecil.

Keterangan:

x : vektor training (x_1, x_2, \dots, x_n)
 T : kategori training vector yang benar untuk pelatihan
 w_j : vektor bobot untuk output unit j ($w_{1j}, \dots, w_{ij}, \dots, w_{nj}$)
 C_j : kategori atau kelas hasil komputasi oleh output unit j
 $\|x - w_j\|$: jarak Euclid antara vektor input dengan (vektor bobot)
 unit output ke j .

Dalam contoh berikut dua vektor referensi dan vektor input yang menunjukkan dua kelas, yaitu 1 dan 2 [13]:

Vektor	Kelas
(1, 1, 0, 0)	1
(0, 0, 0, 1)	2
(0, 0, 1, 1)	2
(1, 0, 0, 0)	1
(0, 1, 1, 0)	2

Dua vektor pertama digunakan sebagai inisialisasi dua vektor referensi (bobot awal). Kemudian output pertama menunjukkan kelas 1 dan output kedua menunjukkan kelas 2 (disimbolkan dengan $C_1 = 1$ dan $C_2 = 2$). Vektor training yang digunakan adalah (0, 0, 1, 1), (1, 0, 0, 0) dan (0, 1, 1, 0). Iterasi yang dilakukan hanya satu kali (epoch = 1), perhitungan LVQ ditunjukkan sebagai berikut:

Step 0 : Inisialisasi vektor referensi

$$W_1 = (1, 1, 0, 0)$$

$$W_2 = (0, 0, 0, 1)$$

Inisialisasi learning rate, $\alpha(0)=0.1$.

Step 1 : bila kondisi STOP belum terpenuhi, kerjakan Step 2-6

Step 2 : untuk input vektor $x = (0, 0, 1, 1)$ dengan $T=2$ kerjakan step 3-4

Step3: kelas 1,

$$j = \sqrt{(0-1)^2 + (0-1)^2 + (1-0)^2 + (1-0)^2} = 2$$

kelas 2,

$$j = \sqrt{(0-0)^2 + (0-0)^2 + (1-0)^2 + (1-1)^2} = 1$$

$$j_{\min} = \min(2, 1) = 1$$

j_{\min} berada pada kelas 2 sehingga $J=2$

Step 4: Karena $T=2$ dan $C_2=2$, update bobot W_2 .

$$w_2 = (0, 0, 0, 1) + 0.1[(0, 0, 1, 1) - (0, 0, 0, 1)]$$

$$= (0, 0, 0.1, 1)$$

Step 2 : untuk input vektor $x = (1, 0, 0, 0)$ dengan $T=1$ kerjakan step 3-4

Step3: kelas 1,

$$j = \sqrt{(1-1)^2 + (0-1)^2 + (0-0)^2 + (0-0)^2} = 1$$

kelas 2,

$$j = \sqrt{(1-0)^2 + (0-0)^2 + (0-0)^2 + (0-1)^2} = 1.41$$

$$j_{\min} = \min(1, 1.41) = 1$$

j_{\min} berada pada kelas 1 sehingga $J=1$

Step 4: Karena $T=1$ dan $C_1=1$, update bobot W_1 .

$$w_1 = (1, 1, 0, 0) + 0.1[(1, 0, 0, 0) - (1, 1, 0, 0)] \\ = (1, 0.9, 0, 0)$$

Step 2 : untuk input vektor $x = (0, 1, 1, 0)$ dengan $T=2$ kerjakan step 3-4

Step3: kelas 1,

$$j = \sqrt{(0-1)^2 + (1-1)^2 + (1-0)^2 + (0-0)^2} = 1.41$$

kelas 2,

$$j = \sqrt{(0-0)^2 + (1-0)^2 + (1-0)^2 + (0-1)^2} = 1.73$$

$$j_{\min} = \min(1.41, 1.73) = 1$$

j_{\min} berada pada kelas 1 sehingga $J=1$

Step 4: Karena $T=2$ tapi $C_1=1$, update bobot W_1 .

$$w_1 = (1, 0.9, 0, 0) - 0.1[(0, 1, 1, 0) - (1, 0.9, 0, 0)] \\ = (1.1, 0.89, -0.1, 0)$$

Step 5 : Reduksi *learning rate* (*alfa*)

Step 6 : Tes kondisi STOP

BAB III

METODE PENELITIAN

Pada bab ini dijelaskan objek dan aspek dalam Tugas Akhir, peralatan, serta langkah-langkah yang digunakan dalam penyusunan Tugas Akhir. Disamping itu, dijelaskan pula prosedur dan proses pelaksanaan tiap-tiap langkah yang dilakukan dalam menyelesaikan Tugas Akhir.

3.1 Objek dan Aspek

Objek yang akan diteliti adalah citra iris mata individu. Sedangkan aspek yang diteliti yaitu mengenali individu melalui identifikasi iris mata dengan menggunakan metode PCA pada ekstraksi ciri dan LVQ pada pencocokan.

3.2 Peralatan

Peralatan utama yang digunakan dalam menyelesaikan Tugas Akhir ini berupa perangkat keras dan perangkat lunak yaitu sebagai berikut :

1. Perangkat keras berupa *Personal Computer* (PC) dengan spesifikasi :
 - *Windows 7 Ultimate 32-bit*
 - Processor Intel® Core™ i3-2375M CPU @1.50GHz
 - Resolusi 1366 x 768 pixel
 - RAM 2GB DDR3
2. Perangkat Lunak
Perangkat lunak yang digunakan untuk membuat simulasi pengenalan individu melalui iris mata adalah matlab dengan versi minimum R2010a.

3.3 Tahap Penelitian

Pada sub bab ini dijelaskan mengenai tahapan-tahapan yang dilakukan dalam menyelesaikan Tugas Akhir ini. Gambar 3.1 merupakan gambaran secara ringkas tahapan yang dijelaskan sebagai berikut:

1. Studi Literatur

Studi Literatur ini dilakukan untuk identifikasi permasalahan dengan mencari referensi yang menunjang penelitian yang berupa Tugas Akhir, jurnal internasional, buku, maupun artikel yang berhubungan dengan topik Tugas Akhir ini.

2. Pengumpulan Data

Pengumpulan data merupakan tahap untuk mengumpulkan data yang diperlukan dalam Tugas Akhir ini. Data yang digunakan dalam Tugas Akhir ini adalah data berupa citra mata yang diambil dari database citra mata *Chinese Academy of Science Institut of Automation (CASIA)*. Database ini diakses melalui website <http://biometrics.idealtest.org/> pada tanggal 27 Maret 2017. Data yang telah terkumpul digunakan sebagai masukan (*input*) dari sistem.

3. Perancangan Perangkat Lunak

Pada tahap ini dilakukan perancangan sistem untuk membuat perangkat lunak yang dapat mengenali individu melalui identifikasi iris mata yang digambarkan pada Gambar 3.2. Secara umum perancangan perangkat lunak yang dibuat dalam Tugas Akhir ini sehingga dapat digunakan untuk identifikasi iris mata adalah sebagai berikut:

a. Akuisisi citra mata

Database citra mata yang telah dikumpulkan kemudian disimpan dalam *disk* komputer sebagai masukan dari sistem.

b. Pra-pengolahan citra iris mata

Sebelum dilakukan proses pengenalan, tahapan yang dilakukan terlebih dahulu pra-pengolahan citra iris mata dimana pada tahap ini dilakukan pemisahan antara bagian yang merupakan iris mata dengan bagian-bagian yang

bukan merupakan iris mata. Pada tahap pra-pengolahan, tahapan yang dilakukan dibagi menjadi dua tahap, yakni tahap lokalisasi iris mata dan tahap normalisasi iris mata. tahap lokalisasi iris mata terdiri dari deteksi tepi canny, transformasi *hough* dan *daugman rubber sheet model*. tahap ini dijelaskan lebih detail pada bab selanjutnya.

c. Ekstraksi ciri

Ekstraksi ciri adalah proses untuk menentukan suatu nilai atau vektor yang digunakan sebagai penciri suatu objek. Metode yang digunakan untuk mendapatkan ciri dari citra iris mata ini adalah PCA.

d. Klasifikasi

Pada tahap ini data citra iris mata dibagi menjadi data *training* dan data *testing*. Kemudian dilakukan proses *taining* untuk mendapatkan nilai bobot dan proses *testing* untuk membandingkan dan mencari jarak terpendek antara data masukan dengan kelas yang ada sehingga didapatkan hasil identifikasi. Metode yang digunakan adalah metode LVQ.

4. Implementasi

Pada tahap implementasi dibuat sistem perangkat lunak dari perancangan yang sudah dilakukan sebelumnya agar mudah dipahami oleh pengguna (*user*). Implementasi tersebut dibuat dengan menggunakan aplikasi Matlab R2013a.

5. Pengujian

Setelah dibuat implementasi, akan dilakukan pengujian untuk memeriksa apakah hasil implementasi sudah sesuai atau masih terjadi *error*.

6. Kesimpulan dan Saran

Setelah dilakukan analisis dan pembahasan maka dapat ditarik suatu kesimpulan dan saran sebagai masukan untuk pengembangan penelitian lebih lanjut.

7. Penyusunan Laporan Tugas Akhir

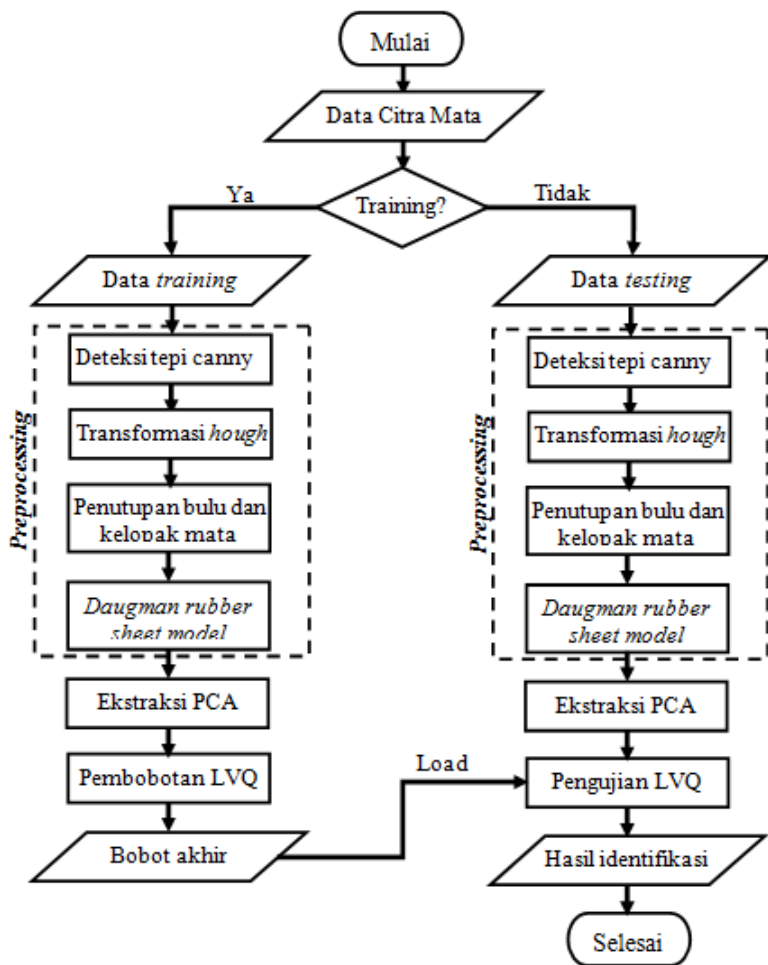
Setelah semua proses selesai dilakukan maka tahap terakhir adalah penyusunan laporan Tugas Akhir.

3.4 Diagram Alir Penelitian

Berdasarkan uraian tersebut diatas, penelitian Tugas Akhir ini dapat dinyatakan dalam diagram alir sebagai berikut.



Gambar 3.1 Diagram alir tahap penelitian



Gambar 3.2 Diagram rancangan perangkat lunak

BAB IV

PERANCANGAN DAN IMPLEMENTASI PERANGKAT LUNAK

Pada bab ini diuraikan mengenai perancangan proses dan algoritma dari sistem yang dibangun. Pembahasan dalam bab ini meliputi perancangan data, perancangan proses dan perancangan antarmuka.

4.1 Perancangan Data

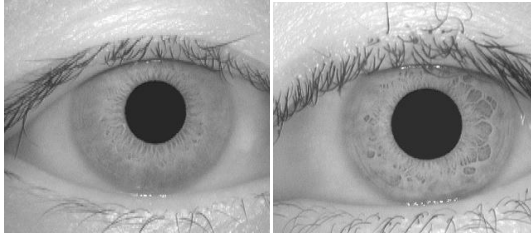
Data yang digunakan dalam proses pengenalan iris mata dibagi menjadi tiga, yaitu data masukan, data proses dan data keluaran. Data masukan merupakan data yang digunakan sebagai masukan dari sistem. Selanjutnya terdapat data proses, yaitu data yang diproses dalam sistem yang nantinya dapat digunakan untuk pengenalan. Data terakhir adalah data keluaran yang merupakan data hasil dari data proses dan data ini dikeluarkan sebagai data informasi.

4.1.1 Data Masukan

Data masukan adalah data yang digunakan sebagai masukan dari sistem. Data masukan yang digunakan dalam Tugas Akhir ini diambil dari database *Chinese Academy of Sciences Institute of Automation (CASIA)*. Database yang digunakan adalah database citra mata versi 1.0 yang memiliki format ekstensi *.bmp dan citra bertipe *grayscale*. Pada Tugas Akhir ini digunakan database citra mata dari 108 mata yang berbeda. Masing-masing mata dianggap sebagai satu kelas. Dalam satu kelas tersebut terdapat 7 citra mata. Gambar 4.1 adalah contoh citra mata dari database CASIA versi 1.0.

Database citra dibagi menjadi dua, yaitu sebagai data *training* dan data *testing*. Data *training* merupakan data yang digunakan untuk melatih sistem. Sedangkan data *testing* adalah data yang digunakan untuk menguji sistem yang hasilnya nanti akan dibandingkan dengan hasil yang diperoleh dari proses pelatihan. Dari tiap kelas didalam database akan dibagi menjadi

dua bagian, yaitu 5 citra sebagai data *training* dan 2 citra sebagai data *testing*.



Gambar 4.1 Contoh citra mata database *CASIA* versi *1.0*

Dalam database *CASIA* versi *1.0* pada masing-masing individu, pengambilan citra dilakukan dalam dua sesi yang mana pada sesi pertama sebanyak 3 citra dan pada sesi kedua sebanyak 4 citra. Contoh pemberian nama pada data base citra mata *CASIA* versi *1.0* adalah sebagai berikut:

“001_1_1”

Keterangan:

001 : citra mata dari individu ke-1

1 : citra mata diambil pada sesi pertama

1 : citra mata diambil pada urutan pertama

“001_2_1”

Keterangan:

001 : citra mata dari individu ke-1

2 : citra mata diambil pada sesi kedua

1 : citra mata diambil pada urutan pertama

4.1.2 Data Proses

Data proses merupakan data yang diproses untuk selanjutnya dapat digunakan untuk pengenalan. Data ini meliputi nama data, tipe data dan keterangan. Data proses dapat dilihat pada Tabel 4.1.

Tabel 4.1 Data proses identifikasi iris mata

No	Nama Data	Tipe Data	Keterangan
1.	Citra mata grayscale	<i>Bitmap</i>	Data citra yang berasal dari database yang bertipe <i>bitmap</i>
2.	Data citra implementasi	<i>double</i>	Data citra yang dikonversi dari tipe <i>bitmap</i> ke data betipe <i>double</i>
3.	Data fitur	<i>double</i>	Vektor fitur yang terdeteksi dan disimpan dalam bentuk *.mat
4.	Data Pencocokan	<i>double</i>	Hasil konversi data fitur bertipe double

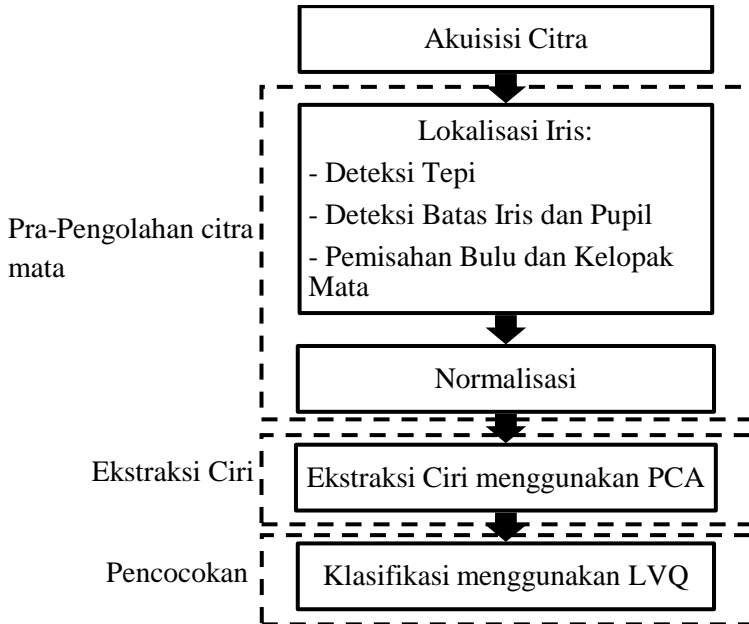
4.1.3 Data Keluaran

Data keluaran merupakan hasil dari data proses yang memberikan informasi berupa letak kelas dari citra tersebut. Selain itu, data keluaran yang dihasilkan adalah berupa citra hasil deteksi batas pupil dan iris, citra hasil pemisahan bulu mata dan kelopak mata serta citra hasil normalisasi.

4.2 Perancangan Proses Pengenalan iris Mata

Dalam perancangan pengenalan iris mata, terdapat beberapa tahapan yang harus dilakukan. Secara garis besar tahapan yang dilakukan adalah akuisisi citra, pra-pengolahan, ekstraksi ciri dan pencocokan atau klasifikasi. Pada tahap pra-pengolahan, tahapan yang dilakukan dibagi menjadi dua, yaitu tahap pertama lokalisasi iris dimana pada tahap ini dilakukan deteksi tepi, pencarian terhadap batas iris dan pupil serta menandai bagian noise dari citra. Tahap pra-pengolahan yang kedua adalah normalisasi citra dimana pada tahap ini dilakukan perubahan bentuk iris yang berupa lingkaran menjadi bentuk iris yang berupa persegi panjang. Selanjutnya citra yang telah ternormalisir di ekstraksi cirinya menggunakan metode PCA. Setelah fitur dari proses ekstraksi didapatkan langkah selanjutnya

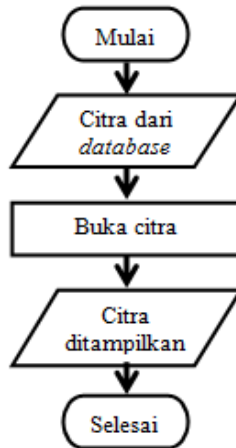
adalah proses pencocokan menggunakan metode LVQ. Gambar 4.2 menunjukkan diagram blok proses pengenalan iris dengan blok pra-pengolahan, ekstraksi ciri dan pencocokan digambarkan dengan garis putus-putus. Tahap-tahap yang lebih detail dijelaskan pada subbab berikutnya.



Gambar 4.2 Diagram blok proses pengenalan iris mata

4.2.1 Akuisisi Citra Mata

Tahap awal sebelum melakukan pengenalan citra adalah tahap akuisi citra. Citra mata yang telah dikumpulkan diasumsikan telah tersimpan dalam *disk* komputer. Seperti pada penjelasan sebelumnya database yang digunakan adalah database citra mata *CASIA* versi *1.0*. Diagram alir akuisisi citra mata dapat dilihat pada Gambar 4.3.

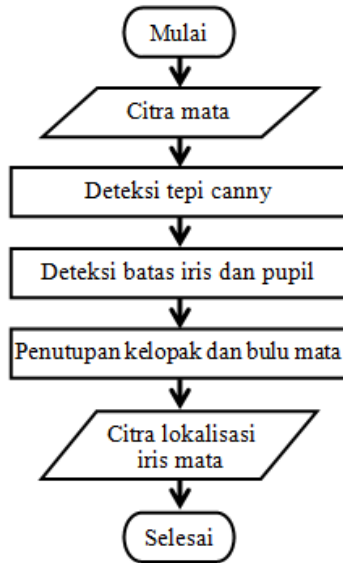


Gambar 4.3 Diagram alir akuisisi citra mata

Untuk menampilkan citra yang diinginkan, buka citra yang tersimpan dalam *disk* komputer. Kemudian citra yang telah dipilih ditampilkan dalam *frame* yang ada di matlab.

4.2.2 Lokalisasi Iris Mata

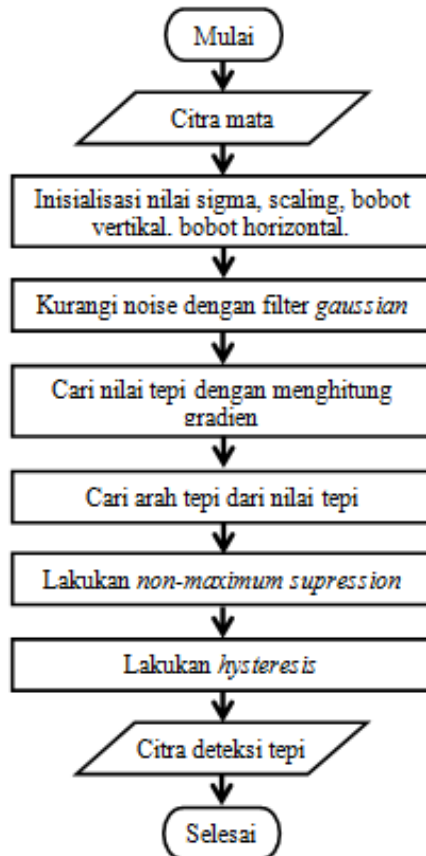
Sebelum melakukan pengenalan individu melalui identifikasi iris mata, diperlukan proses lokalisasi iris yang mana bagian-bagian mata yang tidak diperlukan dalam proses pengenalan harus dipisahkan terlebih dahulu. Tahap pertama dalam lokalisasi iris adalah melakukan deteksi tepi kemudian dilakukan pencarian terhadap batas antara iris mata dengan *sclera* dan batas antara iris mata dengan pupil. Tahap akhir dari proses lokalisasi iris mata adalah menutup bagian yang dianggap sebagai noise seperti bulu mata dan kelopak mata. Gambaran secara umum proses lokalisasi iris mata dapat dilihat pada Gambar 4.4. Tahapan lokalisasi iris secara detail dibahas pada subbab selanjutnya.



Gambar 4.4 Diagram alir proses lokalisasi iris mata

4.2.2.1 Deteksi Tepi Canny

Tahap awal dalam Lokalisasi iris mata adalah deteksi tepi. Pada bab sebelumnya telah dijelaskan bahwa deteksi tepi berfungsi untuk mengidentifikasi garis batas (*boundary*) dari suatu obyek yang terdapat pada citra. Dalam Tugas Akhir ini deteksi tepi yang digunakan adalah deteksi tepi canny. Langkah pertama adalah mengurangi noise pada citra dengan filter *gaussian*. Kemudian dilanjutkan dengan mencari nilai tepi dengan menghitung gradien sehingga arah gradien juga dapat ditentukan. Langkah selanjutnya adalah melakukan proses *non-maximum supression* untuk menelusuri tepi yang ada didalam arah tepi dan meredam nilai pixel. Langkah terakhir adalah proses *hysteresis* yang digunakan untuk menghilangkan garis-garis yang seperti terputus-putus pada tepi obyek. Hasil dari tahap ini adalah citra biner yang menandakan tepi-tepi pada citra asli. Gambar 4.5 menunjukkan proses deteksi tepi.



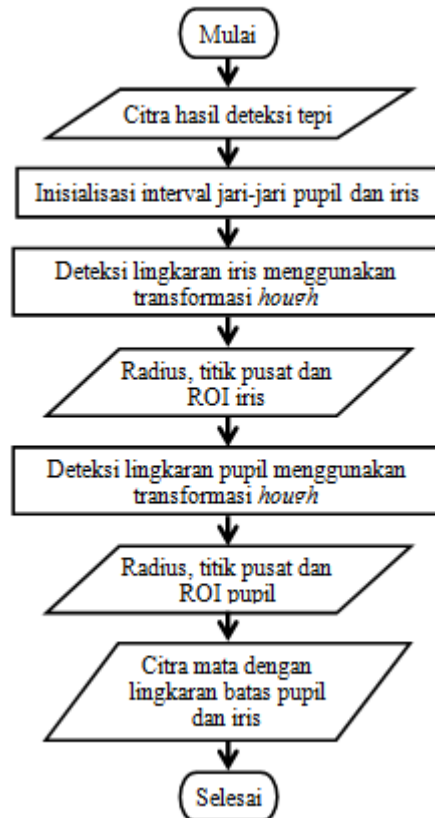
Gambar 4.5 Diagram alir proses deteksi tepi canny

4.2.2.2 Deteksi Batas Pupil dan Iris Mata

Tahap kedua yang dilakukan pada lokalisasi iris mata adalah dengan mendeteksi batas antara iris dan sclera dan batas antara iris dan pupil. Untuk dapat mendeteksi batas-batas tersebut maka digunakan metode transformasi *hough* lingkaran. Pada deteksi batas pupil dan iris transformasi *hough* dilakukan sebanyak dua kali untuk mempercepat komputasi. Sebab apabila transformasi *hough* dilakukan pada masing-masing deteksi

lingkaran iris dan pupil secara bersamaan maka komputasi yang dijalankan akan membutuhkan waktu yang sangat lama. transformasi *hough* yang pertama dilakukan untuk mendapatkan lingkaran iris dan menghasilkan *region of interest* iris. Selanjutnya dilakukan transformasi *hough* yang kedua pada *region of interest* iris untuk mendapatkan lingkaran pupil.

Untuk mendapatkan titik pusat lingkaran iris secara tepat dibutuhkan parameter radius terlebih dahulu. Radius minimum dan maksimum pada databasse citra mata CASIA yaitu 80 piksel dan 150 piksel. Sehingga transformasi *hough* dilakukan secara iterasi dari radius 80 piksel sampai 150 piksel. Jadi, terdapat 70 radius pada bidang akumulator. Radius lingkaran iris yang dipilih adalah radius yang berkaitan dengan titik pusat yang ditemukan. Hasil keluaran dari transformasi *hough* yang pertama adalah radius, titik pusat dan ROI (*Region of Interest*) iris. Transformasi *hough* yang kedua dilakukan pada *region of interest* iris untuk mendapatkan lingkaran pupil. Radius minimum dan maksimum yang digunakan adalah 28 piksel dan 75 piksel. Hasil akhir dari transformasi *hough* adalah radius, titik pusat, dan ROI (*Region of Interest*) pupil. Gambar 4.6 menunjukkan diagram alir proses untuk mendapatkan batas lingkatar iris dan pupil.

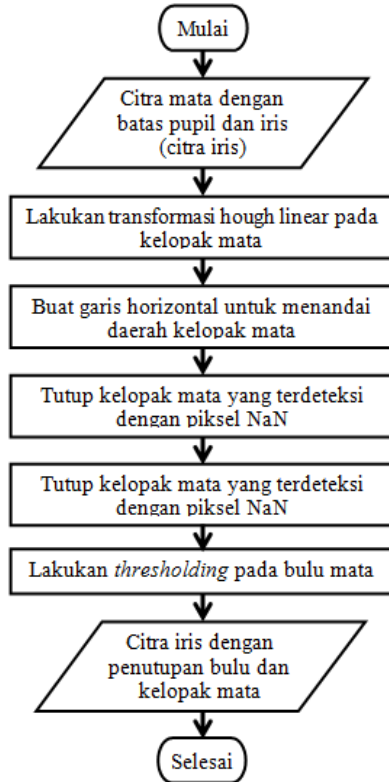


Gambar 4.6 Diagram proses deteksi batas iris dan pupil

4.2.2.3 Penutupan Bulu Mata dan Kelopak Mata

Tahap terakhir dari lokalisasi iris adalah penutupan bagian yang dianggap noise pada citra seperti bulu mata dan kelopak mata. Pada tahap ini dilakukan pendeteksian kelopak mata bagian atas dan bagian bawah dengan transformasi *hough* linier. Selanjutnya dibuat garis horizontal untuk menandai daerah pemisahan. Kemudian setelah terdeteksi, bagian kelopak mata ditutup dengan nilai piksel NaN dan bulu mata yang terlalu banyak menutupi iris mata dideteksi dengan *thresholding*.

Thresholding ini dilakukan pada piksel yang mempunyai intensitas yang hampir serupa dengan bulu mata tersebut. Sehingga saat *thresholding* dilakukan, semua piksel yang nilainya dibawah *threshold* akan berubah. Perubahan piksel ini juga digantikan dengan NaN. Gambar 4.7 menunjukkan diagram alir proses pemisahan kelopak mata dan bulu mata.

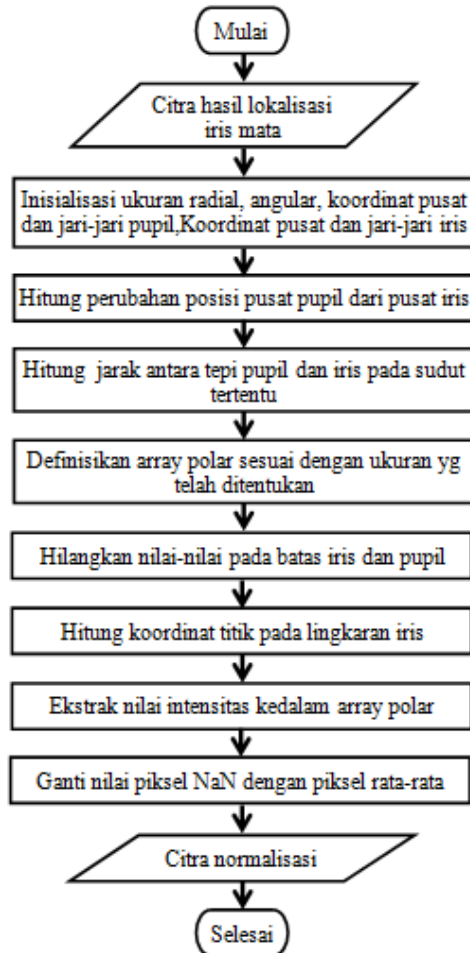


Gambar 4.7 Diagram proses penutupan bulu dan kelopak mata

4.2.3 Normalisasi Iris Mata

Citra mata hasil lokalisasi yang memiliki perbedaan luasan pada daerah iris karena dilatasi selanjutnya dinormalisasi agar mempunyai ukuran yang sama sehingga mudah untuk diekstraksi.

Proses normalisasi ini dilakukan dengan mengubah bentuk lingkaran hasil lokalisasi iris menjadi bentuk persegi panjang yang berukuran 32×256 piksel dengan cara mengubah koordinat kartesian kedalam koordinat polar. Kemudian piksel-piksel yang bernilai NaN diubah dengan nilai piksel rata-rata citra. Gambar 4.8 menunjukkan proses normalisasi citra.



Gambar 4.8 Diagram proses normalisasi citra iris mata

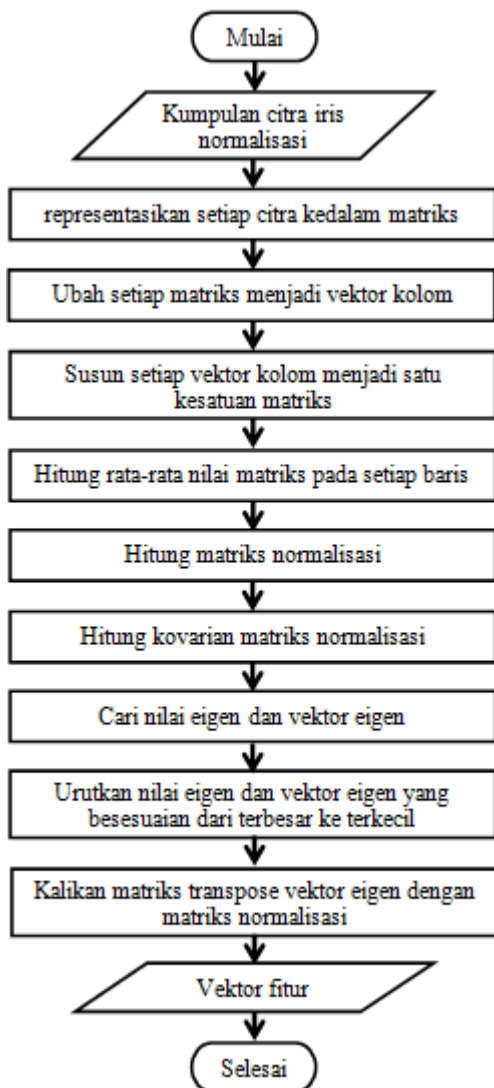
4.2.4 Ekstraksi Ciri

Citra yang telah ternormalisir diekstraksi cirinya menggunakan metode PCA. Pada proses ini dihasilkan vektor fitur yang nantinya digunakan untuk proses klasifikasi. Gambaran ekstraksi ciri dengan metode PCA dapat dilihat pada Gambar 4.9. Langkah awal pada proses ini adalah dengan menyusun matriks $m \times n$ pada setiap citra menjadi vektor kolom $mn \times 1$. Kemudian dicari nilai rata-rata dari matriks. Setelah didapatkan nilai rata-rata dari matriks selanjutnya adalah menghitung matriks normalisasi dengan cara menghitung selisih nilai matriks awal dengan nilai rata-rata citra. Kemudian hitung kovarian matriks dengan mengalikan matriks normalisasi dan tranposenya. Dari kovarian matriks dicari nilai eigen dan vektor eigennya. Selanjutnya memilih beberapa nilai eigen terbesar dan vektor eigen yang bersesuaian dengan nilai eigen yang dipilih sebagai *principal component*. Langkah terakhir adalah mendapatkan vektor fitur dengan mengalikan matriks normalisasi dengan vektor eigen yang diperoleh.

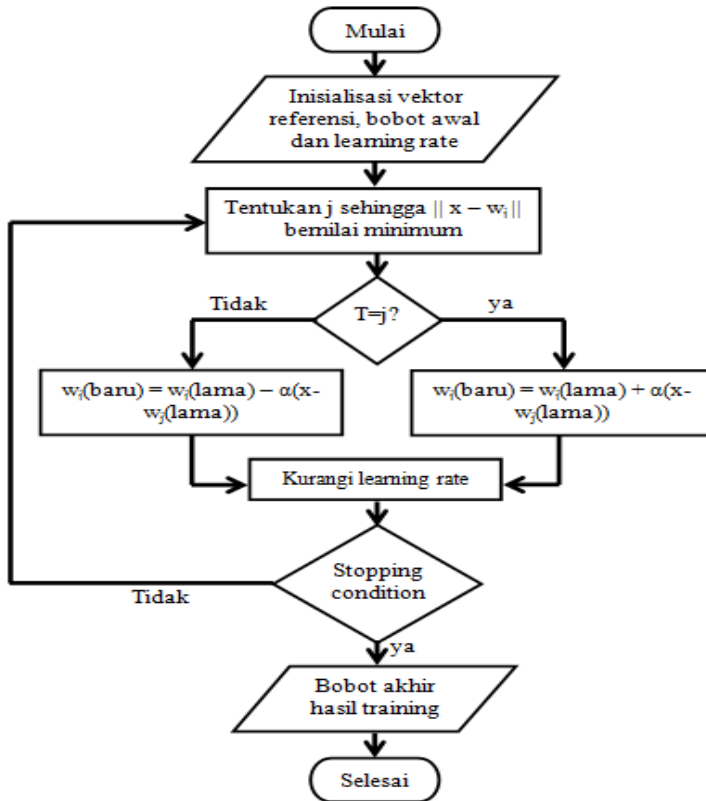
4.2.5 Pencocokan Iris

Pada tahap pencocokan iris, metode yang digunakan adalah LVQ. Dalam metode ini terdapat 2 proses yang dilakukan, yaitu proses pelatihan (*training*) dan pengujian (*testing*). Citra yang telah diperoleh dari hasil ekstraksi ciri kemudian digunakan untuk proses pelatihan. Berikut ini adalah proses perancangan dari metode LVQ untuk proses *training* dan *testing* yang digambarkan pada Gambar 4.10 dan 4.11.

Pada proses *training* pencarian bobot dilakukan sesuai algoritma yang telah dijelaskan pada bab sebelumnya. Tahap awal adalah menginisialisasi bobot awal dan vektor referensi yang didapatkan dari proses ekstraksi ciri. Tahap selanjutnya yang dilakukan adalah menentukan nilai j minimum menghitung $\|x - w_j\|$ dan memperbarui nilai bobot. Kurangi *learning rate*, ulangi tahapan sampai memenuhi kondisi berhenti.

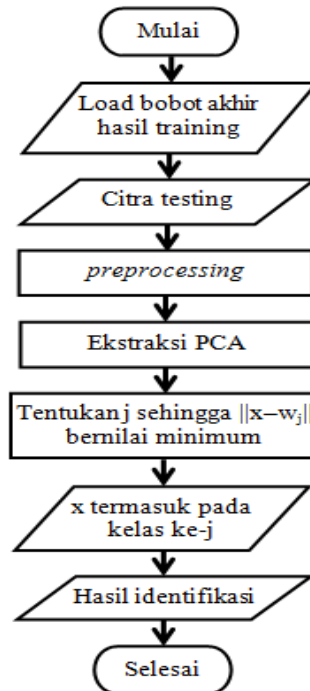


Gambar 4.9 Diagram alir ekstraksi ciri menggunakan PCA



Gambar 4.10 Diagram proses training LVQ

Sedangkan pada proses *testing*, tahapan yang dilakukan adalah mengambil bobot akhir hasil dari proses training dan menginputkan citra testing yang terlebih dahulu melalui tahap *preprocessing* dan ekstraksi ciri. Kemudian menentukan jarak minimum j dengan menghitung $\|x - w_j\|$ dimana x merupakan nilai input dari ekstraksi ciri dan w_j merupakan bobot akhir dari proses *training*, jika x berada pada jarak minimum j maka x termasuk pada kelas j . Hasil akhir proses pengujian ini adalah hasil identifikasi atau kelas dari citra tersebut.



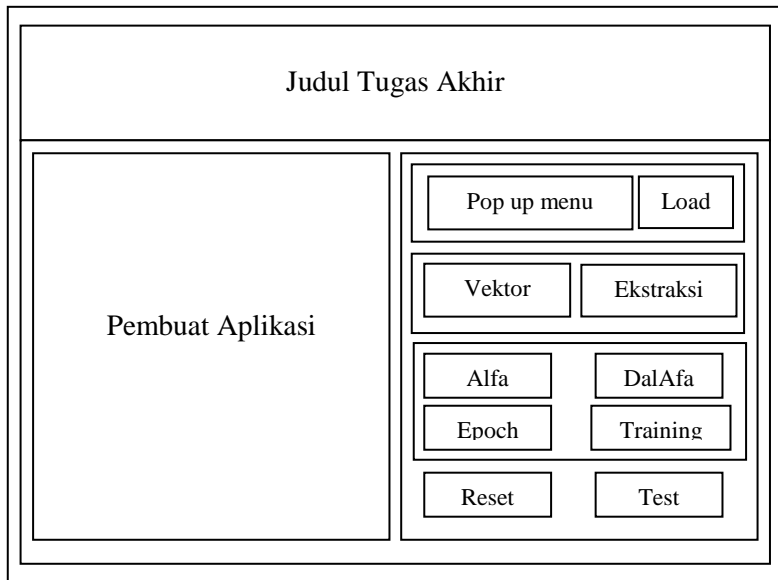
Gambar 4.11 Diagram alir proses testing LVQ

4.3 Perancangan Antarmuka

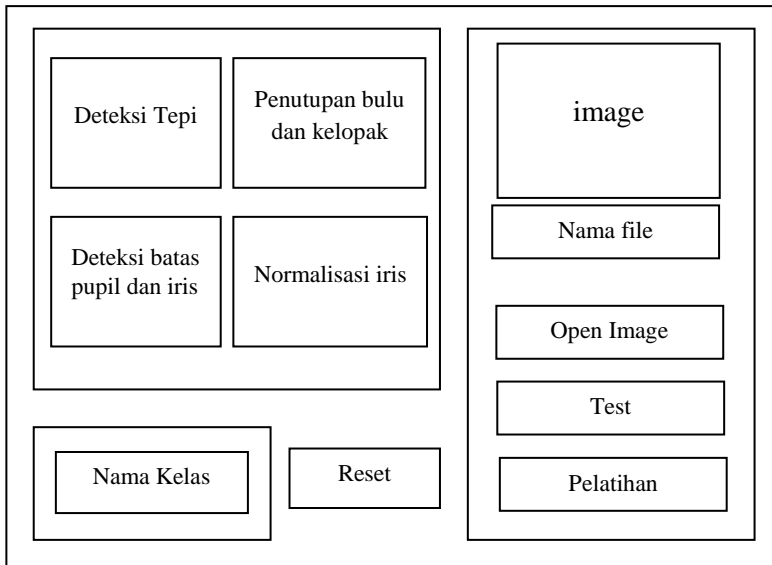
Salah satu aspek penting dalam pembuatan perangkat lunak adalah perancangan antarmuka. Perancangan antarmuka yang baik adalah yang sebanding dengan tingkat *user friendly* sebuah perangkat lunak yang berarti antarmuka didesain semudah dan sesederhana mungkin agar mudah dipahami dan mudah digunakan oleh pemakai. Dalam Tugas Akhir ini terdapat dua antarmuka yang dibuat, yaitu antarmuka halaman utama untuk proses pelatihan (*training*) dan antarmuka untuk proses pengujian (*testing*). Gambar 4.12 dan Gambar 4.13 secara berurutan memperlihatkan rancangan desain antarmuka yang digunakan untuk implementasi. Pada antarmuka halaman utama terdapat panel judul Tugas Akhir, penulis, dan panel load data *training*.

Dalam panel load data *training* terdapat *popup menu* untuk memilih jumlah data, tombol *load* untuk proses prapengolahan, tombol ekstraksi untuk mengekstraksi dari hasil pra-pengolahan, tombol *training* untuk melatih vektor-vektor yang didapat dari hasil ekstraksi ciri, tombol reset untuk menghapus serta tombol *test* untuk menuju *frame* pengujian.

Pada antarmuka pengujian (*testing*) terdapat dua panel. Panel pertama terdapat tombol reset, *text area* yang menunjukkan kelas citra, panel preprocessing yang menampilkan hasil deteksi tepi canny, deteksi batas pupil dan iris, penutupan kelopak dan bulu mata, serta hasil normalisasi. Sedangkan pada panel kedua terdapat tombol open image, tombol test, dan tombol pelatihan untuk kembali ke antarmuka pelatihan serta menampilkan citra yang awal yang akan diidentifikasi dan teks area untuk nama *file* dari citra.



Gambar 4.12 Rancangan antarmuka halaman utama



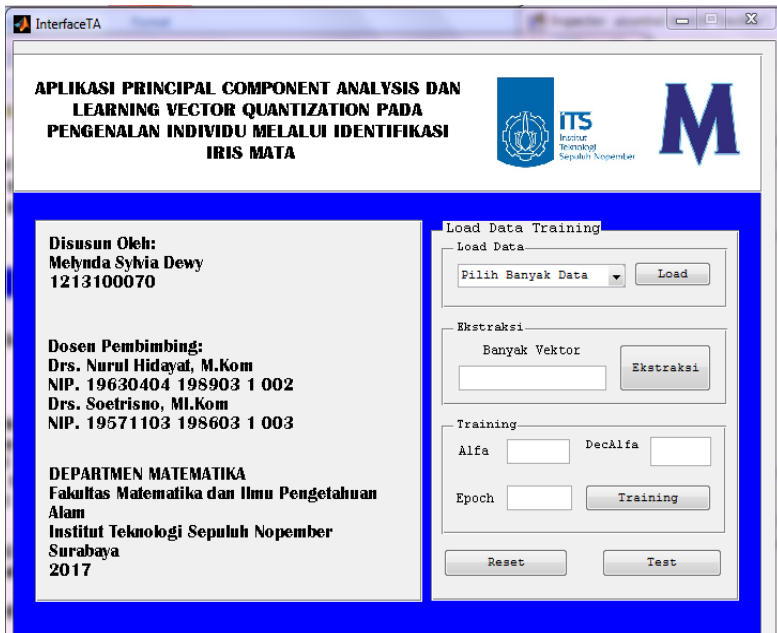
Gambar 4.13 Rancangan antarmuka pengujian

4.4 Implementasi Perangkat Lunak

Setelah perancangan proses dibuat, selanjutnya adalah mengimplementasikan perancangan tersebut kedalam bahasa pemrograman Matlab sehingga sistem dapat dijalankan.

4.4.1 Implementasi Antarmuka

Gambar 4.14 merupakan implementasi antarmuka halaman utama pada pengenalan iris mata, dimana pada antarmuka ini terdapat panel load data, ekstraksi, pelatihan (*training*) dan tombol reset untuk menghapus seluruh nilai yang diinputkan serta tombol test untuk menuju *frame* pengujian.



Gambar 4.14 Antarmuka halaman utama

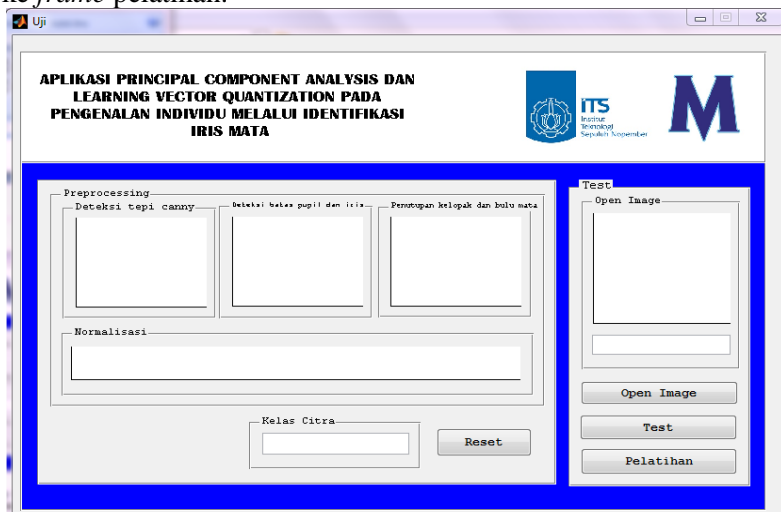
Kegunaan dari setiap *tools* yang berada dalam setiap panel pada antarmuka haaman utama dijelaskan pada tabel 4.2.

Tabel 4.2 Kegunaan *tools* pada setiap panel

Panel	<i>Tools</i>	Kegunaan
Load Data	Pop up menu	Memilih banyak data yang akan di proses.
	Tombol Load	Mendapatkan hasil dari pra-pengolahan citra dari banyak data yang dipilih.
Ekstraksi	<i>Text area</i> (Vektor)	Memasukkan jumlah vektor eigen yang diinginkan
	Tombol Ekstraksi	Melakukan proses ekstraksi ciri dari data yang telah melalui proses pra-pengolahan.

Training	Text area (Alfa)	Memasukkan nilai alfa yang diinginkan
	Text area (Epoch)	Memasukkan nilai epoch yang diinginkan
	Text area (DecAlfa)	Memasukkan nilai target <i>error</i> yang diinginkan
	Tombol Training	Melakukan proses pelatihan dari hasil ekstraksi ciri yang telah dilakukan

Pada gambar 4.15 merupakan antarmuka halaman pengujian (*testing*) terdapat 3 panel dimana pada panel pertama menunjukkan hasil *preprocessing*, kelas citra dan tombol reset untuk . panel kedua menunjukkan hasil kelas citra yang dikenali dan panel ketiga menunjukkan cara untuk mendapatkan citra mata dalam proses pengujian. Dalam panel ketiga terdapat tombol *open file*, *test* dan pelatihan. Tombol *open file* digunakan untuk menampilkan citra yang ingin dikenali, tombol *test* untuk mengetahui hasil dari pengenalan, tombol pelatihan untuk menuju ke *frame* pelatihan.



Gambar 4.15 Antarmuka halaman pengujian

4.4.2 Implementasi Akuisisi Citra Iris Mata

Proses akuisisi citra merupakan proses untuk mendapatkan citra mata yang akan diproses. Proses mendapatkan citra mata yang ada dalam *disk* komputer dapat dilihat pada kode program berikut.

```
[namafile,direktori]=uigetfile({'*.bmp';'*.png';'*.jpg';'*.tif'}, 'Buka Gambar')
if ~isequal(namafile,0);
    handles.data1 =
imread(fullfile(direktori,namafile));
guidata(hObject,handles);
axes(handles.axes5);
imshow(handles.data1);
set(handles.edit1,'String',namafile);
else
return
end
eyeimage=handles.data1;
```

4.4.3 Implementasi Lokalisasi Iris Mata

Implementasi lokalisasi iris mata terdiri dari deteksi tepi canny, deteksi batas pupil dan iris, penutupan kelopak mata dan bulu mata. penjelasan masing-masing implementasi dijelaskan pada subbab selanjutnya.

4.4.3.1 Implementasi Deteksi Tepi

Tahap ini merupakan proses mendapatkan tepian garis dari obyek yang ada dalam citra mata. Deteksi tepi yang digunakan adalah deteksi tepi canny yang telah dimodifikasi. Hal ini dilakukan untuk mengetahui tepi lingkaran pupil dan lingkaran iris. Setelah kedua lingkaran terdeteksi tepinya, bentuk obyek lingkaran akan dikenali dengan transformasi *hough*. Kode program untuk deteksi tepi canny adalah sebagai berikut.

```
[gradient, or] = canny(im, sigma, scaling, vert,
horz);
[I2 or] = canny(image, sigma, scaling, vert, horz);
I3 = adjgamma(I2, 1.9);
I4 = nonmaxsup(I3, or, 1.5);
edgeimage = hysthresh(I4, hithres, lowthres);
```

4.4.3.2 Implementasi Deteksi Batas Pupil dan Iris

Deteksi batas pupil dan iris adalah proses pendeteksian untuk mendapatkan bagian tepi dari iris dan sclera serta bagian tepi dari iris dan pupil. Deteksi ini menggunakan transformasi *hough* lingkaran. Kode program untuk mendeteksi batas pupil dan iris adalah sebagai berikut.

```
% mencari lingkaran iris
[row, col, r] = findcircle(eyeimage, lirisradius,
    uirisradius, scaling, 2, 0.20, 0.19, 1.00, 0.00);

circleiris = [row col r];

% mendeteksi lingkaran iris
imagepupil = eyeimage( irl:iru,icl:icu);

% mencari lingkaran pupil
[rowp, colp, rp] = findcircle(imagepupil,
    lpupilradius, upupilradius
    ,0.6,2,0.25,0.25,1.00,1.00);

rowp = double(rowp);
colp = double(colp);
rp = double(rp);

row = double(irl) + rowp;
col = double(icl) + colp;

row = round(row);
col = round(col);

circlepupil = [row col rp];
```

4.4.3.3 Implementasi Penutupan Kelopak dan Bulu Mata

Pada tahap ini bagian yang dianggap noise seperti kelopak mata dan bulu mata dihilangkan. Langkah awal adalah mencari bagian kelopak mata kemudian menandai bagian kelopak dengan garis horizontal menggunakan transformasi *hough* linier. Setelah mengetahui batas antara kelopak dengan iris mata, selanjutnya adalah memberikan nilai NaN pada kelopak mata dan bulu mata, sehingga nilai intensitas pada titik ini tidak disalah artikan sebagai

daerah iris mata. Kode program untuk menutup kelopak dan bulu mata adalah sebagai berikut.

```
%mencari kelopak mata bagian atas
topeyelid = imagepupil(1:(rowp-r),:);
lines = findline(topeyelid);
if size(lines,1) > 0
    [xl yl] = linecoords(lines, size(topeyelid));
    yl = double(yl) + irl-1;
    xl = double(xl) + icl-1;
    yla = max(yl);
    y2 = 1:yla;
    ind3 = sub2ind(size(eyeimage),yl,xl);
    imagewithnoise(ind3) = NaN;
    imagewithnoise(y2, xl) = NaN;
end

%mencari kelopak mata bagian bawah
bottomeyelid =
imagepupil((rowp+r):size(imagepupil,1),:);
lines = findline(bottomeyelid);
if size(lines,1) > 0
    [xl yl] = linecoords(lines,
size(bottomeyelid));
    yl = double(yl)+ irl+rowp+r-2;
    xl = double(xl) + icl-1;
    yla = min(yl);
    y2 = yla:size(eyeimage,1);
    ind4 = sub2ind(size(eyeimage),yl,xl);
    imagewithnoise(ind4) = NaN;
    imagewithnoise(y2, xl) = NaN;
end

%tanda kelopak mata dengan NaN
ref = eyeimage < 100;
coords = find(ref==1);
imagewithnoise(coords) = NaN;
```

4.4.4 Implementasi Normalisasi Iris Mata

Proses normalisasi iris mata merupakan proses mentransformasikan citra iris mata yang telah terdeteksi dari bentuk lingkaran dan terdeteksi bulu dan kelopak mata menjadi bentuk persegi panjang. Hasil proses normalisasi ini nantinya

digunakan untuk ekstraksi ciri. Kode program proses normalisasi iris mata adalah sebagai berikut.

```
radiuspixels = radpixels + 2;
angledivisions = angulardiv-1;
r = 0:(radiuspixels-1);
theta = 0:2*pi/angledivisions:2*pi;

x_iris = double(x_iris);
y_iris = double(y_iris);
r_iris = double(r_iris);

x_pupil = double(x_pupil);
y_pupil = double(y_pupil);
r_pupil = double(r_pupil);

% hitung perpindahan pusat pupil dari pusat iris
ox = x_pupil - x_iris;
oy = y_pupil - y_iris;

if ox <= 0
    sgn = -1;
elseif ox > 0
    sgn = 1;
end

if ox==0 && oy > 0
    sgn = 1;
end
r = double(r);
theta = double(theta);
a = ones(1,angledivisions+1)*(ox^2 + oy^2);
if ox == 0
    phi = pi/2;
else
    phi = atan(oy/ox);
end
b = sgn.*cos(pi - phi - theta);

% Hitung jari-jari sekitar iris sebagai fungsi
sudut
r = (sqrt(a).*b)+(sqrt(a.*(b.^2)-(a-(r_iris^2)))));
r = r - r_pupil;
```

```

rmat = ones(1,radiuspixels)*r;
rmat = rmat.*
(ones(angledivisions+1,1)*[0:1/(radiuspixels-
1):1]);
rmat = rmat + r_pupil;
% Hilangkan nilai pada inner dan outer boundary
rmat = rmat(2:(radiuspixels-1),:);

% hitung lokasi cartesian dari tiap titik data
disekitar daerah iris
xcosmat = ones(radiuspixels-2,1)*cos(theta);
xsinmat = ones(radiuspixels-2,1)*sin(theta);
xo = rmat.*xcosmat;
yo = rmat.*xsinmat;
xo = x_pupil+xo;
yo = y_pupil-yo;
[x,y] = meshgrid(1:size(image,2),1:size(image,1));
polar_array = interp2(x,y,image,xo,yo);

%ganti nilai piksel NaN ke nilai rata-rata dari
piksel iris
coords = find(isnan(polar_array));
polar_array2 = polar_array;
polar_array2(coords) = 0.5;
avg = sum(sum(polar_array2)) /
(size(polar_array,1)*size(polar_array,2));
polar_array(coords) = avg;

```

4.4.5 Implementasi Ekstraksi Ciri

Pada proses ekstraksi ciri menggunakan metode PCA, setiap citra hasil normalisasi pada masing-masing individu diubah kedalam vektor kolom kemudian disusun kedalam satu matriks agar dapat dicari ciri dari citra yang tersusun. Proses ekstraksi ciri menghasilkan vektor fitur. Vektor ini kemudiandigunakan sebagai bobot dalam proses pembelajaran (*training*) pada tahap klasifikasi. Berikut adalah kode program untuk ekstraksi ciri dengan metode PCA.

```

h = citra; %matriks kumpulan vektor kolom
mean_citra = mean(h,2); %rata-rata matriks
%matrks center/normalisasi
zeromeanI = h-repmat(mean_citra, 1, size(h,2));

```

```

%kovarian matriks dari matriks center
covariance = (1/(size(h,2)-1))*zeromeanI*zeromeanI';
%mencari vektor eigen dan nilai eigen dari matriks
kovarian
[eigVec eigVal]=eig(covariance);

%mengurutkan nilai eigen dari terbesar ke kecil dan
vektor eigen yang bersesuaian
for j = 1:size(eigVal,1)
pcaEigenVals(j,j)=eigVal(index(j,:),index(j,:));
    pcaEigenVecs(:,j)= eigVec(:,index(j,:));
end
%memilih sejumlah vektor eigen
evecs = pcaEigenVecs(:, 1:num_eigen);
%vektor fitur
features = evecs'*zeromeanI;

```

4.4.6 Implementasi Pencocokan

Pada tahap ini dilakukan pelatihan (*training*) dan pengujian (*testing*). Kode program proses *training* adalah sebagai berikut.

```

epoch = 1;
while (epoch<=maxEpoch) && (alfa>=minAlfa)
for k = 1:baris
for c = 1:baris1
        D(c) = 0;
for i = 1:kolom1
D(c) = D(c)+(input(k,i)-w(c,i))^2;
end
        D(c) = sqrt(D(c));
end
        [jmin, idx] = min(D);
if idx==target(k)
        w(idx,:) = w(idx,:)+alfa*(input(k,:)-
w(idx,:));
else
        w(idx,:) = w(idx,:)-alfa*(input(k,:)-
w(idx,:));
end
end
        alfa = alfa-decAlfa*alfa;
        epoch = epoch+1;
end

```

Kode program untuk proses *testing* adalah sebagai berikut.

```
bobot = w;  
H = [];  
[baris, kolom] = size(input);  
[baris1, kolom1] = size(bobot);  
for k = 1:baris  
    for c = 1:baris1  
        D(c) = 0;  
        for i = 1:kolom  
            D(c) = D(c) + (input(k,i) - w(c,i))^2;  
        end  
        D(c) = sqrt(D(c));  
    end  
    [jmin, idx] = min(D);  
    H = [H; idx];  
end
```

BAB V

UJI COBA DAN EVALUASI

Pada bab ini dijelaskan hasil dari ujicoba berdasarkan implementasi sistem yang telah dibuat.

5.1 Data Uji Coba

Data yang digunakan pada Tugas Akhir ini adalah data citra mata keabuan yang diambil dari database *CASIA* versi 1.0. Dari database ini diambil sebanyak 108 mata berbeda dengan 7 citra pada masing-masing mata. Pada setiap mata citra dibagi menjadi data *training* dan data *testing* seperti yang telah dijelaskan pada bab sebelumnya. Dalam Tugas Akhir ini dilakukan pengujian sebanyak empat kali, yakni menggunakan 30, 50, 80 dan 108 mata.

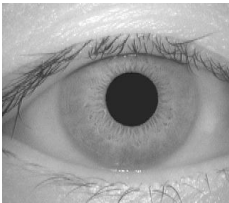

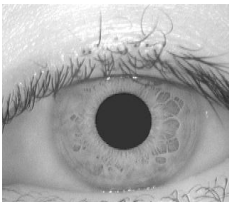
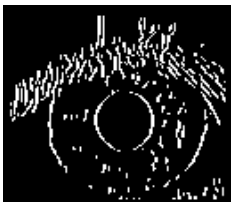
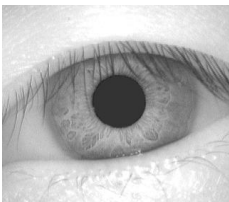

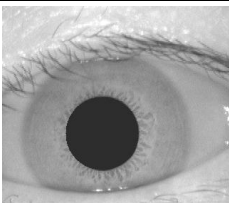
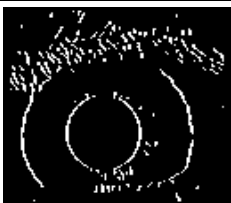
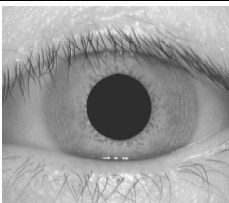
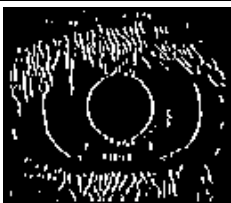
5.2 Uji Coba Proses Lokalisasi Iris Mata

Uji coba dilakukan pada sampel sepuluh citra mata. Pengambilan ini disebabkan karena terlalu banyak melibatkan citra. Kemudian masing-masing sampel diujikan pada tiap tahap lokalisasi iris mata. Proses pada lokalisasi iris mata ini meliputi deteksi tepi canny, deteksi batas pupil dan iris, dan penutupan kelopak mata dan bulu mata. masing-masing uji coba dijelaskan pada subbab selanjutnya.

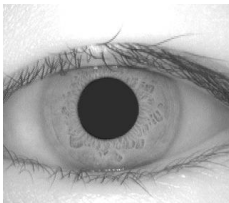

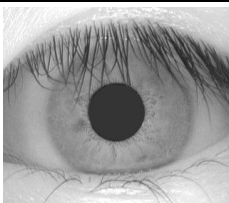
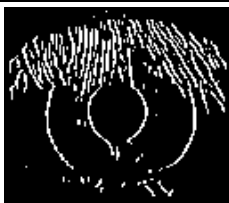
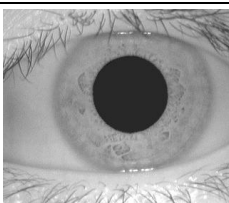
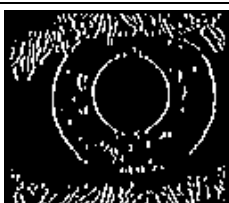
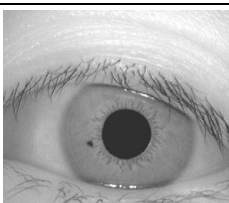

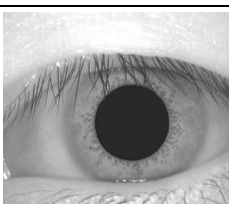
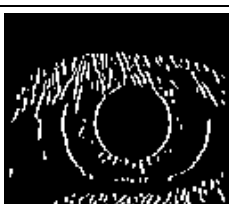
5.2.1 Uji Coba Deteksi Tepi

Hasil uji coba deteksi tepi pada sepuluh citra dapat dilihat pada Tabel 5.1 dan Tabel 5.2. Sampel citra mata merupakan representasi dari 108 citra mata individu dalam database. Citra awal dan hasil deteksi tepi ditunjukkan pada kolom ketiga dan keempat. Dari Tabel 5.1 dan Tabel 5.2 dapat dilihat bahwa hasil dari deteksi tepi canny dapat menunjukkan dengan jelas tepian pada pupil, iris serta bulu mata. Hal ini jelas akan sangat membantu dalam tahap deteksi batas pupil dan iris dengan transformasi *hough* lingkaran.

Tabel 5.1 Hasil uji coba deteksi tepi canny

No	Nama File	Citra Asli	Deteksi Tepi Canny
1.	001_1_1.bmp		
2.	003_2_2.bmp		
3.	007_1_2.bmp		
4.	013_2_1.bmp		
5.	015_2_2.bmp		

Tabel 5.2 Lanjutan hasil uji coba deteksi tepi canny

No	Nama File	Citra Asli	Deteksi Tepi Canny
6.	020_2_1.bmp		
7.	023_1_1.bmp		
8.	025_1_2.bmp		
9.	032_2_2.bmp		
10.	039_1_1.bmp		

5.2.2 Uji Coba Deteksi Batas Pupil dan Iris

Hasil dari deteksi tepi canny kemudian digunakan pada pendeteksian batas pupil dan iris. Seperti yang dijelaskan pada bab sebelumnya, pendeteksian batas pupil dan iris menggunakan transformasi *hough*. Tabel 5.3 dan Tabel 5.4 menunjukkan hasil pengujian dari deteksi batas pupil dan iris. Dari tabel tersebut dapat dilihat bahwa transformasi *hough* mampu mendeteksi batas antara *sclera* dan iris serta batas antara pupil dan iris dengan baik. Namun pada beberapa citra masih terdapat kegagalan dalam pendeteksian. Hal yang menyebabkan kegagalan tersebut adalah batas daerah iris yang terdeteksi terlalu sempit karena adanya noise yang terlalu banyak atau daerah iris yang terlalu jauh karena deteksi tepian iris terlalu kecil.

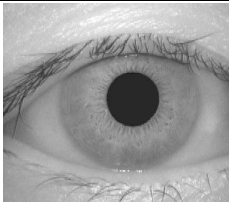
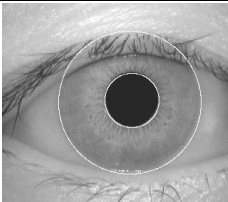
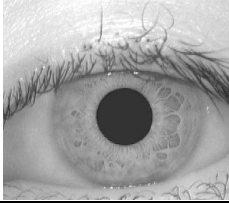
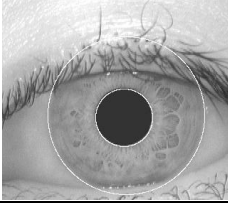
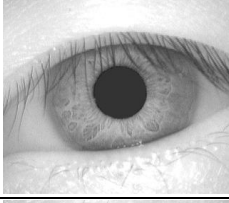
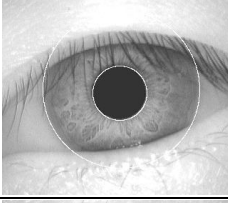
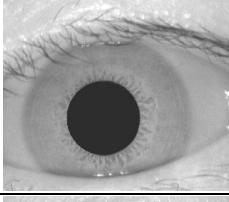
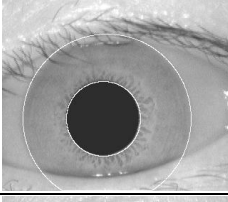
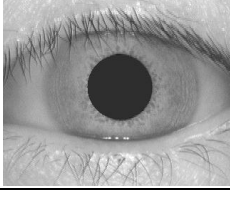
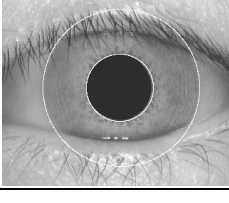
5.2.3 Uji Coba Penutupan Bulu dan Kelopak Mata

Pada proses ini dilakukan pendeteksian noise kemudian dilakukan penutupan pada noise atau memberikan nilai piksel NaN pada noise yang terdeteksi. Tabel 5.5 dan 5.6 menunjukkan hasil penutupan bulu dan kelopak mata. Dari tabel tersebut, bagian hitam merupakan penanda atau nilai piksel NaN yang menutupi bagian bulu dan kelopak mata. Namun pada beberapa citra, bulu dan kelopak mata tidak tertutup dengan sempurna. Hal ini disebabkan oleh pencahayaan pada saat penangkapan citra.

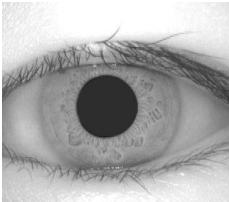
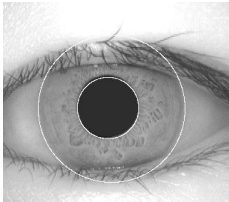
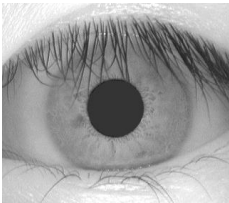
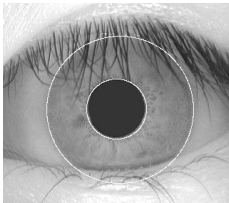
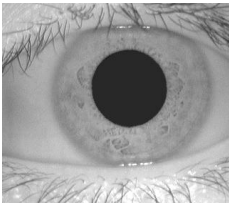
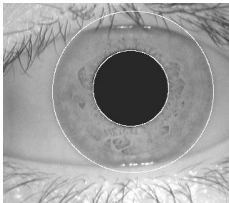
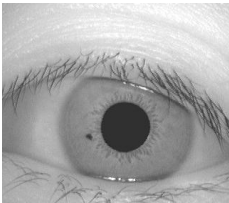
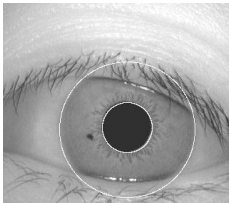
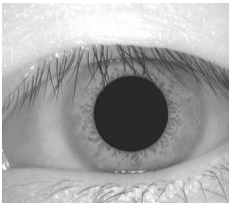
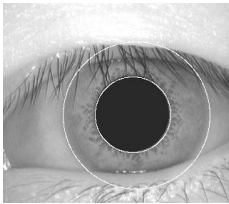
5.3 Uji Coba Normalisasi

Hasil dari normalisasi citra dapat dilihat pada Tabel 5.7 dan Tabel 5.8. Seperti yang telah dijelaskan pada bab sebelumnya. Proses normalisasi citra iris mata merupakan proses mengubah citra mata yang terdeteksi berbentuk lingkaran menjadi citra berbentuk persegi agar mudah diekstraksi cirinya. Ukuran dari citra normalisasi ini adalah 32×256 piksel.

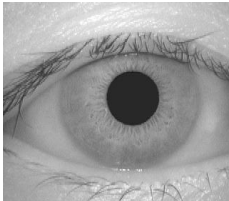
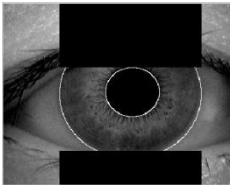
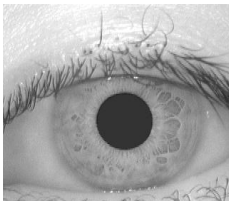
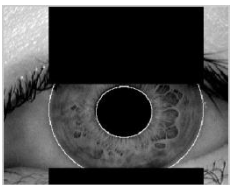
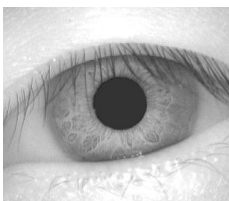

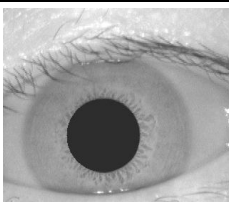
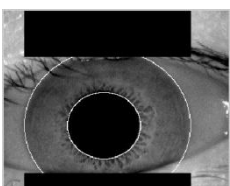
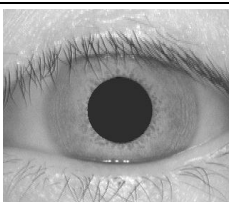
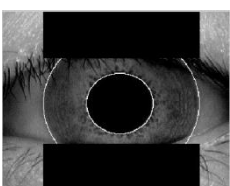
Tabel 5.3 Hasil uji coba deteksi batas pupil dan iris

No	Nama File	Citra Asli	Deteksi Batas Pupil dan Iris
1.	001_1_1.bmp		
2.	003_2_2.bmp		
3.	007_1_2.bmp		
4.	013_2_1.bmp		
5.	015_2_2bmp		

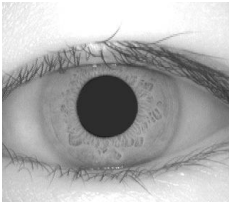
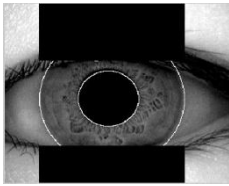
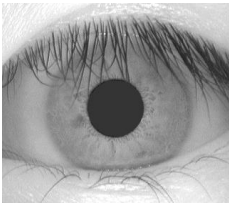
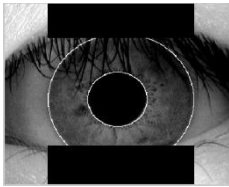
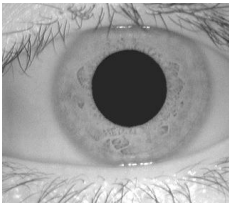
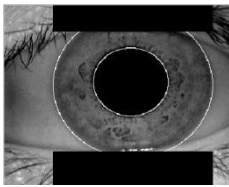
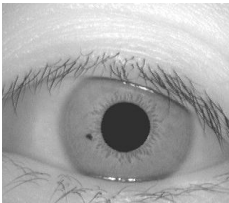
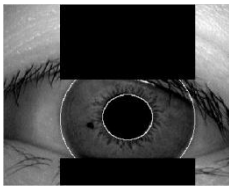
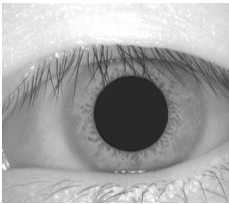
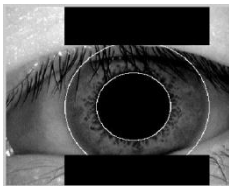
Tabel 5.4 Lanjutan hasil uji coba deteksi batas pupil dan iris

No	Nama File	Citra Asli	Deteksi Batas Pupil dan Iris
6.	020_2_1.bmp		
7.	023_1_1.bmp		
8.	025_1_2.bmp		
9.	032_2_2.bmp		
10.	039_1_1.bmp		

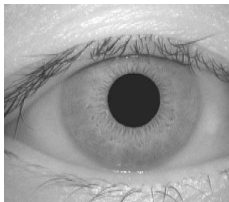

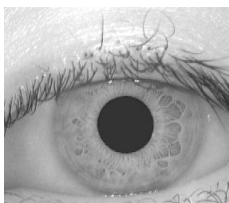

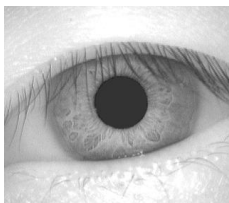

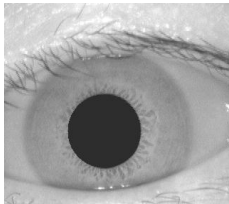

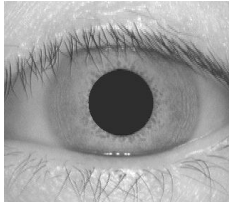

Tabel 5.5 Hasil uji coba penutupan bulu dan kelopak mata

No	Nama File	Citra Asli	Penutupan Bulu dan Kelopak Mata
1.	001_1_1.bmp		
2.	003_2_2.bmp		
3.	007_1_2.bmp		
4.	013_2_1.bmp		
5.	015_2_2.bmp		

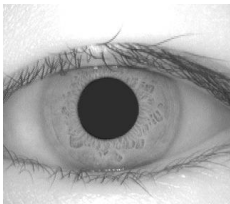

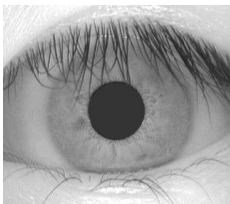

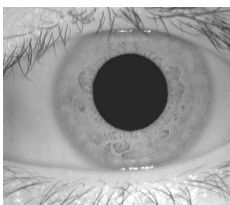
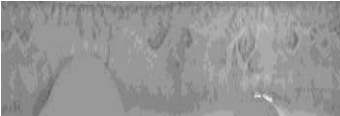
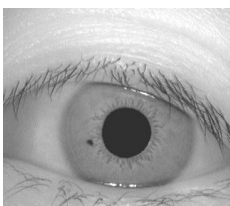

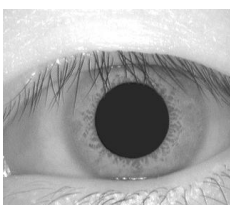
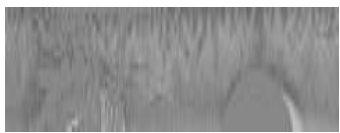
Tabel 5.6 Lanjutan hasil uji coba penutupan bulu dan kelopak mata

No	Nama File	Citra Asli	Penutupan Bulu dan Kelopak Mata
6.	020_2_1.bmp		
7.	023_1_1.bmp		
8.	025_1_2.bmp		
9.	032_2_2.bmp		
10.	039_1_1.bmp		

Tabel 5.7 Hasil uji coba normalisasi

No	Nama File	Citra Asli	Normalisasi
1.	001_1_1.bmp		
2.	003_2_2.bmp		
3.	007_1_2.bmp		
4.	013_2_1.bmp		
5.	015_2_2.bmp		

Tabel 5.8 Lanjutan hasil uji coba normalisasi

No	Nama File	Citra Asli	Normalisasi
6.	020_2_1.bmp		
7.	023_1_1.bmp		
8.	025_1_2.bmp		
9.	032_2_2.bmp		
10.	039_1_1.bmp		

5.4 Uji Coba Ekstraksi Ciri

Citra hasil normalisasi kemudian diolah untuk di ekstraksi cirinya. Citra normalisasi ini dikonversi kedalam nilai double sehingga didapatkan nilai matriks pada setiap citra. Nilai matriks pada setiap citra kemudian diubah menjadi vektor kolom dan disusun menjadi satu kesatuan matriks agar didapatkan vektor fitur dari citra-citra yang telah terkumpul. Vektor fitur yang didapatkan merupakan wakil dari setiap citra. Tabel 5.9 merupakan contoh vektor fitur dari 10 vektor eigen.

Tabel 5.9 Contoh hasil vektor fitur dengan 10 vektor eigen

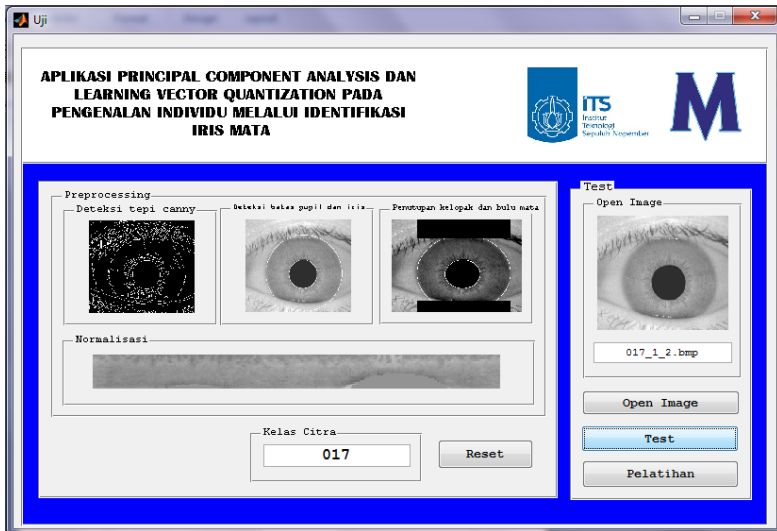
Fitur 1	Fitur 2	Fitur 3	Fitur 4	Fitur ...	Fitur ...
-1.886	2.230	0.881	-3.691
0.376	-0.462	-0.265	-0.345
-0.025	1.475	0.269	0.547
0.396	0.383	0.812	0.088
0.229	-0.532	0.058	-0.499
-0.309	0.355	0.479	-0.049
-0.259	0.288	-0.769	-0.353
-0.505	-0.406	-0.599	0.531
-0.241	0.146	0.047	0.222
0.183	-0.032	-0.033	0.015

Hasil vektor fitur yang didapatkan kemudian disimpan yang nantinya akan dijadikan *input* pada proses *training* dalam metode LVQ. Pengambilan nilai eigen mempengaruhi akurasi pada proses pengenalan iris mata. Dalam pengujian, dilakukan pengambilan vektor eigen sebanyak 5, 15, 25, 40, dan 50.

5.5 Uji Coba Pencocokan

Sebelum dilakukan pengenalan atau pengujian, pada tahap ini terlebih dahulu dilakukan pelatihan jaringan dimana nilai bobot didapatkan dari vektor fitur hasil dari proses ekstraksi ciri. Perhitungan bobot akhir dalam Tugas Akhir ini menggunakan learning rate (*alfa*) sebesar 0.1, iterasi (*epoch*) yang dilakukan sebanyak 100 iterasi dan nilai *error* sebesar 0.1. Setelah

didapatkan nilai bobot akhir pada proses pelatihan langkah selanjutnya adalah dilakukan pengujian yang mana nilai bobot pada proses pengujian ini dicari jarak terdekatnya dengan nilai bobot yang didapatkan pada proses pelatihan. Gambar 5.1 dan Tabel 5.10 menunjukkan contoh hasil uji coba pada pencocokan.



Gambar 5.1 Hasil uji coba pencocokan

Dari Gambar 5.1 tersebut dapat dilihat bahwa hasil uji coba pada citra mata '017_1_2.bmp' berhasil diidentifikasi yang mana hasil kelas citra menunjukkan kelas yang sama, yaitu pada kelas 017 sehingga citra mata tersebut dikenali. Pada Tabel 5.10 dapat dilihat bahwa citra mata yang diuji berhasil dikenali. Namun, ada beberapa citra mata yang tidak berhasil dikenali. Penyebab kegagalan dari pengenalan dibahas pada subbab selanjutnya.

Tabel 5.10 Contoh hasil pengenalan metode LVQ

No	Nama Citra	Dikenali pada kelas	Keterangan
1.	001_1_1.bmp	001	Benar
2.	003_1_3.bmp	003	Benar
3.	007_2_4.bmp	015	Salah
4.	011_1_3.bmp	004	Salah
5.	015_1_2.bmp	005	Salah
6.	018_2_3.bmp	018	Benar
7.	020_2_1.bmp	020	Benar
8.	023_1_1.bmp	023	Benar
9.	026_1_3.bmp	029	Salah
10.	030_2_4.bmp	030	Benar

5.6 Akurasi Sistem Pengenalan Iris

Setelah dilakukan uji coba pada sistem pengenalan iris mata, akurasi sistem dalam pengenalan iris mata dapat diukur. Presentase pengenalan ini diperoleh dengan membagi jumlah keseluruhan citra yang berhasil dikenali dengan total keseluruhan citra yang diuji. Seperti yang telah dijelaskan pada subbab sebelumnya, uji coba dilakukan pada 30, 50, 80 dan 108 mata dengan pengambilan vektor eigen sebanyak 5, 15, 25, 40, 50, 75, 100, dan 125 pada proses ekstraksi ciri PCA. Sedangkan pada proses pelatihan dengan metode LVQ digunakan *learning rate*, *epoch*, dan penurunan *learning rate* yang mempunyai nilai tetap pada setiap uji coba yang dilakukan. *Learning rate* dan penurunan *learning rate* yang digunakan adalah sebesar 0.1 dan *epoch* sebanyak 100 *epoch*.

Tabel 5.11 menunjukkan hasil akurasi pada pengujian 30 mata yang mana terlihat bahwa akurasi terbaik diperoleh dari pengambilan 75 vektor eigen. Jumlah citra yang berhasil dikenali sebanyak 195 citra dari 210 citra yang di uji. Presentase yang diperoleh adalah sebesar 92.857%.

Tabel 5.11 Hasil akurasi uji coba 30 mata

		Jumlah keber- hasilan	Jumlah gagal dikenali	Total	Akutasi (%)	Total akurasi (%)
5 vektor eigen	Training	102	48	150	68	64.285
	Testing	33	27	60	55	
15 vektor eigen	Training	109	41	150	72.667	70
	Testing	38	22	60	63.333	
25 vektor eigen	Training	116	34	150	77.333	75.238
	Testing	42	18	60	70	
40 vektor eigen	Training	123	27	150	82	82.857
	Testing	51	9	60	85	
50 vektor eigen	Training	138	12	150	92	91.429
	Testing	54	6	60	90	
75 vektor eigen	Training	142	9	150	94.667	92.857
	Testing	53	7	60	88.333	
100 vektor eigen	Training	134	16	150	89.333	88.095
	Testing	51	9	60	85	
125 vektor eigen	Training	133	17	150	88.667	87.619
	Testing	51	10	60	85	

Pada uji coba kedua yang menggunakan 50 mata, jumlah citra yang berhasil dikenali sebanyak 326 citra dari 350 citra dari pengambilan 75 vektor eigen sehingga akurasi yang diperoleh adalah sebesar 93.143%. Rincian akurasi dari uji coba pada 50 mata dapat dilihat pada Tabel 5.12.

Selanjutnya pada uji coba yang ketiga menggunakan 80 mata hasil terbaik diperoleh dari pengambilan 75 vektor eigen dengan 523 citra dikenali dari 560 citra yang menghasilkan akurasi sebesar 93.3928%. Tabel 5.13 menunjukkan hasil akurasi dari uji coba pada 80 mata.

Tabel 5.12 Hasil akurasi uji coba 50 mata

		Jumlah keber- hasilan	Jumlah gagal dikenali	Total	Akutasi (%)	Total akurasi (%)
5 vektor eigen	Training	180	70	250	72	67.714
	Testing	57	43	100	57	
15 vektor eigen	Training	188	62	250	75.2	71.389
	Testing	69	31	100	69	
25 vektor eigen	Training	206	44	250	82.4	79.714
	Testing	73	27	100	73	
40 vektor eigen	Training	221	29	250	86.8	85.428
	Testing	81	19	100	81	
50 vektor eigen	Training	231	19	250	92.4	92
	Testing	84	16	100	84	
75 vektor eigen	Training	238	12	250	95.2	93.143
	Testing	88	12	100	88	
100 vektor eigen	Training	226	24	250	90.4	89.714
	Testing	88	12	100	88	
125 vektor eigen	Training	224	26	250	89.6	88.571
	Testing	86	14	100	86	

Sedangkan pada pengujian terakhir menggunakan 108 mata hasil terbaik diperoleh sebanyak 706 citra dikenali dari citra yang diujikan dengan pengambilan 75 dan 100 vektor eigen. Presentase akurasi yang diperoleh sebesar 93.518%. Hasil akurasi uji coba pada 108 mata ditunjukkan pada Tabel 5.14.

Keseluruhan akurasi hasil uji coba secara ringkas ditunjukkan pada Tabel 5.15 yang memperlihatkan bahwa pada pengujian yang telah dilakukan, akurasi terbaik pada masing-masing uji coba didapatkan dari pengambilan 75 vektor eigen.

Tabel 5.13 Hasil akurasi uji coba 80 mata

		Jumlah keber- hasilan	Jumlah gagal dikenali	Total	Akutasi (%)	Total akurasi (%)
5 vektor eigen	Training	291	109	400	72.75	68.928
	Testing	95	65	160	59.375	
15 vektor eigen	Training	323	77	400	80.75	78.928
	Testing	119	41	160	74.375	
25 vektor eigen	Training	348	52	400	87	86.25
	Testing	130	30	160	81.25	
40 vektor eigen	Training	363	37	400	90.75	88.214
	Testing	131	29	160	81.875	
50 vektor eigen	Training	370	30	400	92.5	90.357
	Testing	136	24	160	85	
75 vektor eigen	Training	377	23	400	94.25	93.393
	Testing	146	14	160	91.25	
100 vektor eigen	Training	364	36	400	91	90.178
	Testing	141	19	160	86.25	
125 vektor eigen	Training	362	38	400	90.5	90
	Testing	142	18	160	86.875	

Tabel 5.14 Hasil uji coba 108 mata

		Jumlah keber- hasilan	Jumlah gagal dikenali	Total	Akutasi (%)	Total akurasi (%)
5 vektor eigen	Training	384	156	540	71.111	68.783
	Testing	136	80	216	62.963	
15 vektor eigen	Training	453	87	540	83.889	80.952
	Testing	159	57	216	73.611	
25 vektor eigen	Training	471	69	540	87.222	85.582
	Testing	176	40	216	81.481	
40 vektor eigen	Training	480	60	540	88.889	87.169
	Testing	179	37	216	82.87	
50 vektor eigen	Training	497	43	540	92.037	90.476
	Testing	187	29	216	86.574	
75 vektor eigen	Training	515	25	540	95.370	93.518
	Testing	192	24	216	88.889	
100 vektor eigen	Training	493	47	540	91.2963	90.079
	Testing	188	28	216	87.037	
125 vektor eigen	Training	489	51	540	90.5556	90.079
	Testing	189	27	216	87.5	

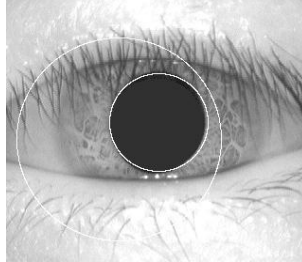
Tabel 5.15 Akurasi uji coba secara keseluruhan

Learning rate (alfa)=0.1, epoch=100, minimum alfa = 0.1				
	30 mata	50 mata	80 mata	108 mata
Akurasi (%) 5 vektor eigen	64.285	67.714	68.928	68.783
Akurasi (%) 15 vektor eigen	70	71.389	78.928	80.952
Akurasi (%) 25 vektor eigen	75.238	79.714	86.25	85.582
Akurasi (%) 40 vektor eigen	82.857	85.428	88.214	87.169
Akurasi (%) 50 vektor eigen	91.429	92	90.357	90.476
Akurasi (%) 75 vektor eigen	92.857	93.143	93.393	93.518
Akurasi (%) 100 vektor eigen	88.095	89.714	90.178	90.079
Akurasi (%) 125 vektor eigen	87.619	88.571	90	90.079

5.7 Penyebab Kegagalan Pengenalan Iris Mata

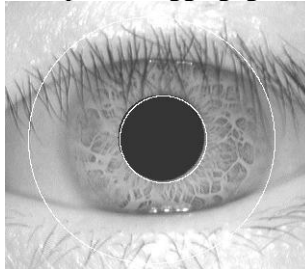
Setelah dilakukan uji coba pada sistem pengenalan iris mata, terdapat beberapa penyebab kegagalan dalam pengenalan iris mata, yaitu:

1. Kegagalan pada saat lokalisasi iris mata. Ada beberapa penyebab yang dapat mengagalkan lokalisasi iris mata, diantaranya adalah sebagai berikut:
 - a. Daerah iris yang tidak terdeteksi pada saat lokalisasi iris. Gagalnya lokalisasi iris ini disebabkan oleh daerah iris mata yang tertutup oleh bulu dan kelopak mata seperti pada Gambar 5.2 sehingga ekstraksi fitur dan pengenalan pada iris mata menjadi tidak optimal.



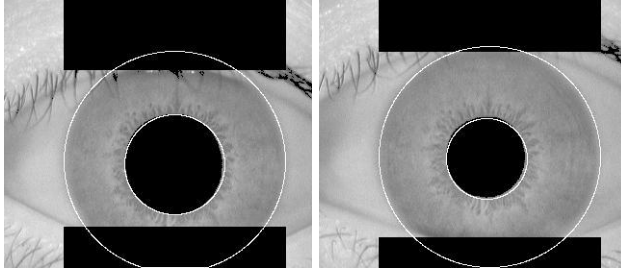
Gambar 5.2 Kegagalan deteksi batas iris karena tertutup oleh bulu mata

- b. Daerah iris yang gagal dideteksi karena batas iris yang terdeteksi jauh dari batas iris yang sebenarnya. Seperti yang ditunjukkan pada Gambar 5.3 batas iris mata yang sebenarnya terlalu tipis sehingga gagal dideteksi.



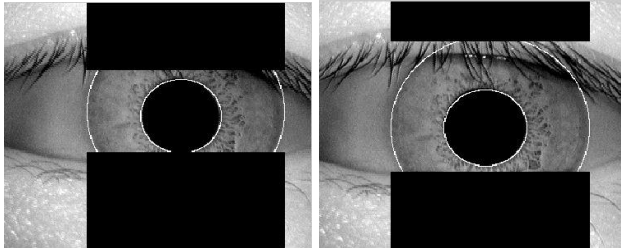
Gambar 5.3 Kegagalan pendeteksian batas luar iris

2. Pupil mengalami dilatasi (membesar atau mengecil) sehingga kepadatan dan kerenggangan daerah iris tidak seragam pada subyek yang sama seperti yang ditunjukkan pada Gambar 5.4.



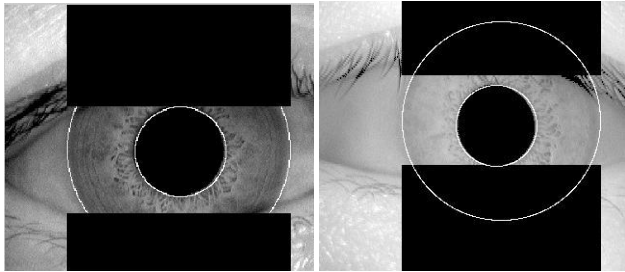
Gambar 5.4 Kondisi citra yang gagal dikenali karena perbedaan besar pupil

3. Pendeteksian daerah bulu dan kelopak mata yang berbeda seperti yang ditunjukkan pada Gambar 5.5



Gambar 5.4 Kondisi citra yang gagal dikenali karena perbedaan pendeteksian bulu dan kelopak mata

4. Daerah yang dianggap sebagai noise seperti bulu dan kelopak mata terlalu banyak sehingga daerah iris yang terdeteksi menjadi terlalu sedikit yang mempengaruhi proses pengenalan, seperti yang terlihat pada Gambar 5.6.



Gambar 5.6 Kondisi daerah iris yang terlalu sedikit

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan analisis terhadap hasil pengujian program, maka dapat diambil kesimpulan sebagai berikut:

1. Metode *Principal Component Analysis* dan *Learning Vector Quantization* telah berhasil diterapkan untuk mengenali iris mata individu dari database *CASIA versi 1.0*.
2. Hasil akurasi dari identifikasi iris mata menggunakan metode PCA dan LVQ terbaik diperoleh sebesar 93.518% dengan pengambilan 75 vektor eigen, *learning rate* sebesar 0.1, penurunan *learning rate* 0.1 dan maksimum *epoch* sebesar 100.
3. Akurasi terbaik dari semua hasil uji coba diperoleh dari pengambilan 75 vektor eigen.

6.2 Saran

Ada beberapa hal yang penulis sarankan untuk pengembangan penelitian selanjutnya:

1. Menambahkan *image enhancement* pada tahap pra-pengolahan agar didapatkan ciri yang lebih baik.
2. Menggabungkan metode PCA dan LVQ dengan metode lain.
3. Mengaplikasikan metode PCA dan LVQ pada obyek biometrik yang lain

DAFTAR PUSTAKA

- [1] Putra, D. (2009). *Sistem Biometrika: Konsep Dasar, Teknik Analisis Citra, dan Tahapan Membangun Aplikasi Sistem Biometrika*. Yogyakarta: Andi.
- [2] Shabrina, M. (2012). *Pengenalan Iris Mata Menggunakan Metode Analisis Komponen Utama (Principal Component Analysis - PCA) Dan Jaringan Syaraf Tiruan Perambatan Balik*. Semarang: Universitas Diponegoro.
- [3] Wildes, R. (1997). *Iris Recognition: An Emerging Biometric Technology*. *PROCEEDINGS OF THE IEEE* , Vol 85 (9).
- [4] Jatra, M. (2011). *Identifikasi Iris Mata Menggunakan Metode Analisis Komponen Utama dan Perhitungan Jarak Euclidean*. Semarang: Universitas Diponegoro.
- [5] Faundra, M. R. (2011). *Aplikasi Filter Log Gabor Pada Sistem Pengenalan Iris Mata*. Tugas Akhir. Institut Teknologi Sepuluh Nopember.
- [6] Nor'aini, A., Rohilah, S., & Azilah, S. (2013). *Classification of Iris Recognition using Principal Component Analysis dan Support Vector Machine*. *IEEE International Conference on Signal and Image Processing Applications (ICSIPA)* , 134-139.
- [7] Buddharpawar, A. S., & Subbaraman, S. (2015). *Iris Recognition based on PCA for Person Identification*. *International Journal of Computer Applications* , 9-12.
- [8] Isnanto, R. R. (2009). *Identifikasi Iris Mata Menggunakan Tapis Gabor Wavelet dan Jaringan Syaraf Tiruan Learning Vector Quantization (LVQ)*. *TEKNIK* , Vol 30 (19-24).
- [9] Anil, K., Shalil, P., & Sharath, P. (2003). *Biometric Recognition: Security and Privacy Concern*. *IEEE Computer Society* , 34-35.
- [10] Putra, D. (2010). *Pengolahan Citra Digital*. Yogyakarta: Andi.

- [11] Libor, M. (2003). ***Recognition of Human Iris Pattern for Biometric Identification***. Retrieved June 10, 2017, from <http://www.csse.uwa.edu.au/pk/student/project/libor>.
- [12] Pau, L. C., & Sumam, A. A. (2012). **Face Recognition Using Principal Component Analysis Method**. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)* , 135-139.
- [13] Fausett, L. (1994). ***Fundamentals of Neural Networks: Architectures, Algorithms and Applications***. Prentice Hall International, Inc.
- [13] <http://biometrics.idealtest.org/>

LAMPIRAN A
Hasil Uji Coba Pada 30 Mata Dengan 75 Vektor Eigen

No	Nama Kelas	Training		Testing	
		Benar	Salah	Benar	Salah
1	001	5	0	2	0
2	002	4	1	2	0
3	003	5	0	2	0
4	004	5	0	2	0
5	005	5	0	2	0
6	006	5	0	2	0
7	007	4	1	1	1
8	008	5	0	2	0
9	009	5	0	2	0
10	010	5	0	2	0
11	011	5	0	2	0
12	012	5	0	1	1
13	013	5	0	2	0
14	014	5	0	2	0
15	015	4	1	2	0
16	016	5	0	2	0
17	017	5	0	2	0
18	018	4	1	1	1
19	019	4	1	2	0
20	020	5	0	1	1
21	021	5	0	2	0
22	022	4	1	1	1
23	023	5	0	2	0
24	024	4	1	1	1
25	025	5	0	2	0
26	026	4	1	1	1
27	027	5	0	2	0
28	028	5	0	2	0
29	029	5	0	2	0
30	030	5	0	2	0
Total		142	8	53	7
Akurasi (%)		94.667		88.333	
Total akurasi (%)		92.857			

LAMPIRAN B
Hasil Uji Coba Pada 50 Mata Dengan 75 Vektor Eigen

No	Nama Kelas	Training		Testing	
		Benar	Salah	Benar	Salah
1	001	5	0	2	0
2	002	5	0	2	0
3	003	5	0	2	0
4	004	5	0	2	0
5	005	5	0	2	0
6	006	5	0	2	0
7	007	4	1	1	1
8	008	5	0	2	0
9	009	5	0	2	0
10	010	5	0	2	0
11	011	4	1	1	1
12	012	5	0	1	1
13	013	5	0	1	1
14	014	5	0	2	0
15	015	5	0	2	0
16	016	5	0	2	0
17	017	5	0	2	0
18	018	5	0	1	1
19	019	4	1	2	0
20	020	5	0	2	0
21	021	5	0	2	0
22	022	5	0	1	1
23	023	5	0	2	0
24	024	5	0	2	0
25	025	5	0	2	0
26	026	4	1	1	1
27	027	5	0	2	0
28	028	5	0	2	0
29	029	4	1	1	1
30	030	5	0	2	0
31	031	5	0	2	0
32	032	5	0	2	0
33	033	4	1	1	1

34	034	5	0	2	0
35	035	5	0	2	0
36	036	5	0	2	0
37	037	4	1	1	1
38	038	5	0	2	0
39	039	5	0	2	0
40	040	5	0	2	0
41	041	4	1	2	0
42	042	4	1	1	1
43	043	4	1	2	0
44	044	5	0	2	0
45	045	4	1	2	0
46	046	4	1	1	1
47	047	5	0	2	0
48	048	5	0	2	0
49	049	5	0	2	0
50	050	5	0	2	0
Total		238	12	88	12
Akurasi (%)		95.2		88	
total akurasi		93.142			

LAMPIRAN C
Hasil Uji Coba Pada 80 Mata Dengan 75 Vektor Eigen

No	Nama Kelas	Training		Testing	
		Benar	Salah	Benar	Salah
1	001	5	0	2	0
2	002	5	0	2	0
3	003	5	0	2	0
4	004	5	0	2	0
5	005	5	0	2	0
6	006	5	0	2	0
7	007	3	2	1	1
8	008	5	0	2	0
9	009	5	0	2	0
10	010	5	0	2	0
11	011	4	1	1	1
12	012	5	0	2	0
13	013	5	0	2	0
14	014	5	0	2	0
15	015	5	0	2	0
16	016	5	0	2	0
17	017	5	0	2	0
18	018	4	1	1	1
19	019	4	1	2	0
20	020	5	0	2	0
21	021	5	0	2	0
22	022	5	0	1	1
23	023	5	0	2	0
24	024	5	0	2	0
25	025	5	0	2	0
26	026	4	1	2	0
27	027	5	0	2	0
28	028	5	0	2	0
29	029	4	1	1	1
30	030	4	1	2	0
31	031	5	0	2	0
32	032	5	0	2	0
33	033	4	1	2	0

34	034	5	0	1	1
35	035	5	0	2	0
36	036	5	0	2	0
37	037	4	1	1	1
38	038	5	0	2	0
39	039	5	0	2	0
40	040	5	0	2	0
41	041	4	1	2	0
42	042	4	1	1	1
43	043	4	1	2	0
44	044	5	0	2	0
45	045	4	1	1	1
46	046	4	1	1	1
47	047	5	0	2	0
48	048	5	0	2	0
49	049	5	0	2	0
50	050	5	0	2	0
51	051	5	0	2	0
52	052	4	1	2	0
53	053	5	0	2	0
54	054	5	0	2	0
55	055	5	0	2	0
56	056	5	0	2	0
57	057	5	0	2	0
58	058	5	0	2	0
59	059	5	0	2	0
60	060	5	0	2	0
61	061	5	0	2	0
62	062	5	0	2	0
63	063	5	0	2	0
64	064	5	0	2	0
65	065	5	0	2	0
66	066	4	1	1	1
67	067	4	1	2	0
68	068	5	0	2	0
69	069	5	0	2	0
70	070	4	1	1	1
71	071	4	1	2	0

72	072	4	1	1	1
73	073	4	1	1	1
74	074	5	0	2	0
75	075	5	0	2	0
76	076	5	0	2	0
77	077	5	0	2	0
78	078	5	0	2	0
79	079	4	1	2	0
80	080	5	0	2	0
Total		377	23	146	14
Akurasi (%)		94.25		91.25	
total akurasi		93.392			

LAMPIRAN D
Hasil Uji Coba Pada 108 Mata Dengan 75 Vektor Eigen

No	Nama Kelas	Training		Testing	
		Benar	Salah	Benar	Salah
1	001	5	0	2	0
2	002	5	0	2	0
3	003	5	0	2	0
4	004	5	0	2	0
5	005	5	0	2	0
6	006	5	0	2	0
7	007	4	1	1	1
8	008	5	0	2	0
9	009	5	0	2	0
10	010	5	0	2	0
11	011	4	1	1	1
12	012	5	0	2	0
13	013	5	0	1	1
14	014	5	0	2	0
15	015	5	0	2	0
16	016	5	0	2	0
17	017	5	0	2	0
18	018	4	1	1	1
19	019	5	0	2	0
20	020	5	0	2	0
21	021	5	0	2	0
22	022	5	0	1	1
23	023	5	0	2	0
24	024	5	0	1	1
25	025	5	0	2	0
26	026	4	1	2	0

27	027	5	0	2	0
28	028	5	0	2	0
29	029	5	0	2	0
30	030	4	1	2	0
31	031	5	0	2	0
32	032	5	0	2	0
33	033	5	0	1	1
34	034	5	0	2	0
35	035	5	0	2	0
36	036	5	0	2	0
37	037	5	0	0	2
38	038	4	1	1	1
39	039	5	0	2	0
40	040	5	0	2	0
41	041	4	1	2	0
42	042	5	0	1	1
43	043	4	1	1	1
44	044	5	0	2	0
45	045	4	1	1	1
46	046	5	0	1	1
47	047	5	0	2	0
48	048	5	0	2	0
49	049	5	0	2	0
50	050	5	0	2	0
51	051	5	0	2	0
52	052	4	1	1	1
53	053	5	0	2	0
54	054	5	0	2	0
55	055	5	0	2	0
56	056	5	0	2	0

57	057	5	0	2	0
58	058	5	0	2	0
59	059	5	0	2	0
60	060	5	0	2	0
61	061	5	0	2	0
62	062	5	0	2	0
63	063	5	0	2	0
64	064	5	0	2	0
65	065	5	0	2	0
66	066	4	1	1	1
67	067	4	1	2	0
68	068	5	0	1	1
69	069	5	0	2	0
70	070	4	1	1	1
71	071	4	1	2	0
72	072	4	1	2	0
73	073	4	1	1	1
74	074	5	0	2	0
75	075	5	0	2	0
76	076	5	0	2	0
77	077	4	1	1	1
78	078	5	0	2	0
79	079	4	1	1	1
80	080	5	0	2	0
81	081	4	1	2	0
82	082	4	1	1	1
83	083	5	0	2	0
84	084	5	0	2	0
85	085	5	0	2	0
86	086	4	1	2	0

87	087	5	0	2	0
88	088	5	0	2	0
89	089	5	0	2	0
90	090	5	0	2	0
91	091	4	1	2	0
92	092	5	0	2	0
93	093	5	0	2	0
94	094	5	0	1	1
95	095	4	1	2	0
96	096	5	0	2	0
97	097	5	0	2	0
98	098	5	0	2	0
99	099	5	0	2	0
100	100	5	0	2	0
101	101	5	0	2	0
102	102	5	0	2	0
103	103	4	1	2	0
104	104	4	1	2	0
105	105	5	0	2	0
106	106	4	1	1	1
107	107	5	0	2	0
108	108	5	0	2	0
Total		514	26	192	24
Akurasi (%)		95.3703		88.8889	
total akurasi		93.518			

BIODATA PENULIS



Bernama lengkap Melynda Sylvia Dewy dengan nama panggilan Melynda. Lahir di Probolinggo, 13 Mei 1995. Anak pertama dari Ayah Wasit Sarwono dan Ibu Surifa. Jenjang pendidikan formal yang ditempuh yaitu SDN Nguling 1 (2001-2007), SMPN 1 Nguling (2007-2010), dan SMAN 1 Grati (2010-2013).

Pada tahun 2013 penulis diterima di Jurusan Matematika ITS melalui jalur tulis SBMPTN untuk menempuh pendidikan S1 selama empat tahun. DiJurusan Matematika ITS penulis mengambil bidang minat ilmu komputer (*Computer Science*). Penulis juga aktif berorganisasi di KM ITS, yaitu sebagai staff Departemen Sains Terapan dan Keprofesian (Sainstek) di Himpunnan Mahasiswa Matematika ITS (2014-2015), Sekretaris Applied Science Department (ASCI) HIMATIKA ITS (2015-2016), Staff Dana Usaha di Lembaga Dakwah Jurusan Ibnu Muqlah Matematika ITS (2014-2015), dan penulis juga aktif dalam kepanitiaan acara tingkat Nasional yaitu Olimpiade Matematika ITS sebagai Crew of Secretarial di dalam kampus.

Jika ingin memberikan saran, kritik dan diskusi mengenai Tugas Akhir ini, dapat menghubungi melalui email melynda.sylvia.dewy@gmail.com. Semoga Tugas Akhir ini dapat bermanfaat.